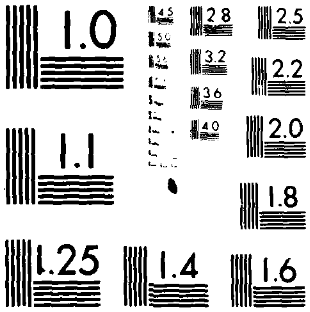
AD-A113 044   CLEMSON UNIV SC DEPT OF MATHEMATICAL SCIENCES   F/G 9/2
BASKET METHOD FOR SELECTING BALANCED SAMPLES. PART III. COMPUTE--ETC(U)
NOV 81 K T WALLENIUS, S L BENZ   N00014-75-C-0451
UNCLASSIFIED N134   NL

1 OF 1
AD
A113044

END
DATE
FILMED
5-82
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

DEPARTMENT OF MATHEMATICAL SCIENCES

CLEMSON UNIVERSITY
Clemson, South Carolina

12

BASKET METHOD FOR SELECTING BALANCED SAMPLES - PART III: COMPUTER SOURCE PROGRAMS

K. T. Wallenius
and
Stephen L. Benz*

Department of Mathematical Sciences

Clemson University

Technical Report #374

November 1981

N134

This work was supported in part by the Office of Naval Research under Congract N00014-75-C-0451.

*Computer Science Department, Clemson University, Clemson, SC 29631.

INTRODUCTION

In this third of a series of three documents describing the basket method of sampling, source computer codes are provided for the convenience of users who may want to modify the programs to meet special local needs. The programming language used is the General Electric Mark III version of FORTRAN. These source programs and associated load modules are available to all users through the GE/USAF Copper Impact Time Sharing System.

The programs can be run using a load module driver named BASKET* with the GE system command:

```
RUN BASKET*
```

Source programs written in IBM FORTRAN IV are also available from the authors.

The source listing of four main programs and their associated subroutines are contained in this document. They are:

BASKET: A driver program which calls the appropriate load modules as specified by the user.

BASKSIM: Program used during the perspective analysis.

BASSEL: Program for basket formation to be used in actual application.

BASKAWARD: Program which prices the unsampled proposed based on the results of negotiating the proposals in the selected basket.

```
BASKET      17:25PDT    08/19/81

10 INTEGER ANS
11 CALL IFBRK(*1)
20 1 PRINT,"WHICH OF THE FOLLOWING WOULD YOU LIKE TO RUN:"
30 PRINT," "
40 PRINT,"  1) BASKET SIMULATION MODEL"
50 PRINT,"  2) BASKET SELECTION MODEL"
60 PRINT,"  3) AWARD MODEL"
61 PRINT,"  4) TERMINATE"
70 PRINT," "
80 PRINT," "
90 PRINT,"INPUT 1,2, OR 3"
100 PRINT," "
110 INPUT,ANS
120 IF(ANS.NE.1.AND.ANS.NE.2.AND.ANS.NE.3.AND.ANS.NE.4)GOTO 1
130 IF(ANS.EQ.1)CALL SYSTEM('/BASKSIM*')
140 IF(ANS.EQ.2)CALL SYSTEM('/BASSEL*')
150 IF(ANS.EQ.3)CALL SYSTEM('/BASAWRD*')
160 IF(ANS.EQ.4)STOP1
170 END
```

```
1 $OPTION PRTP
100C  MAIN PROGRAM
110           CALL DSPLAY
120           CALL INPUT
130        1  CALL PROGRM($2,$3)
140           CALL LIST($1)
150        2  CALL SIMULA($1)
160        3  CALL PRICE($1)
170           STOP
180           END
190C
200C
210           SUBROUTINE DSPLAY
220           WRITE("     ",1)
230         1 FORMAT(' ',76('*')/' *',74X,'*'/' *',25X,'PERSPECTIVE ANALYSIS '
240          &,'OF',26X,'*'/' *',30X,'BASKET METHOD',31X,'*'/' *',12X,'APPLIED '
250          &,'TO HISTORICAL BID/NEGOTIATED RELATIONSHIPS',12X,'*'/' *',74X
260          &,'*',/' *',25X,'SIMULATION AND PRICING',27X,'*'/' *',
270          &25X,'VERSION IV - JULY 1981',27X,'*'/' *',74X
280          &,'*'/' *',14X,'THEORETICAL DEVELOPMENT - DR. K. T. WALLENIUS',
290          &15X,'*'/' *',10X,'ALGORITHM DESIGN AND IMPLEMENTATION -',
300          &' DR. STEPHEN BENZ',10X,'*'/' *',74X,'*'/' *',29X,'CLEMSON',
310          &' UNIVERSITY',28X,'*'/' *',31X,'CLEMSON S.C.',31X,'*'/' *',
320          &74X,'*'/' ',76('*'))
330           RETURN
340           END
350C
360C
370           SUBROUTINE INPUT
380           CALL NUMCHR($1)
390           CALL CHRTYP
400        1  CALL BIDS
410           RETURN
420           END
430C
440           SUBROUTINE NUMCHR(*)
450C   READS NUMBER OF CHARACTER TYPES INTO - NCHAR.
460C
470           COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRBSK(350,12)
480           INTEGER CHRNAM,CHRBSK
490     1     WRITE("     ",2)
491 PRINT,""
500     2     FORMAT(' ',/,' ENTER NUMPER OF CHARACTER TYPES.')
510           READ("     ",*) NCHAR
520           IF(NCHAR.EQ.0) NCHAR=1
530           IF(NCHAR.EQ.1) RETURN1
540           IF(NCHAR.GT.0.AND.NCHAR.LE.12) RETURN
550           WRITE("     ",3) NCHAR
560     3     FORMAT(' ',/,' *** ERROR ***  NUMBER OF CHARACTER TYPES,',I3,',',
570          &/,15X,' IS NOT BETWEEN 1 AND 12 INCLUSIVE.')
580           WRITE("     ",4)
581 PRINT,""
590     4     FORMAT(' DO YOU WISH TO REENTER NUMBER OF CHARACTER TYPES?',
600          &' (YES OR NO)')
610           CALL ANSWER($1)
620           STOP
630           END
640C
```

```
650C
660         SUBROUTINE CHRTYP
670C   READS CHARACTER TYPES INTO - CHRNAM(N,K), K=1,2.
680C
690         COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRBSK(350,12)
700         INTEGER BLANK
710         INTEGER CHRNAM,CHRBSK
720         DATA  BLANK/'    '/
730    1    WRITE("     ",2) NCHAR
731 PRINT,""
740    2    FORMAT(' ',/,' ENTER',I3,', 8 - CHARACTER ID''S FOR CHARACTER',
750        &' TYPES, ONE PER LINE.')
760         DO 3 J=1,NCHAR
770    3    READ ("     ",4) (CHRNAM(J,K),K=1,2)
780    4    FORMAT(2A4)
790         DO 5 N=1,NCHAR
800         IF(CHRNAM(N,1).EQ.BLANK.AND.CHRNAM(N,2).EQ.BLANK) GO TO 6
810    5    CONTINUE
820         RETURN
830    6    WRITE("     ",7) NCHAR
840    7    FORMAT(' ',/,' *** ERROR ***  NUMBER OF CHARACTER TYPES DOES',/,
850        &17X,' NOT EQUAL',I3,', OR DATA NOT ENTERED IN PROPER FORMAT.')
860         WRITE("     ",8)
861 PRINT,""
870    8    FORMAT(' DO YOU WISH TO REENTER CHARACTER TYPES? (YES OR NO)')
880         CALL ANSWER(81)
890         STOP
900         END
910C
920C
930         SUBROUTINE BIDS
940C   READS DATA.
950C          BIDNUM(N) - BID PROPOSAL NUMBER.
960C          BIDPRI(N) - BID PRICE.
970C          BIDNEG(N) - NEGOTIATED PRICE.
980C          BIDTYP(N) - INTEGER VALUES FROM 1 TO 12 CORRESPONDING
990C                      TO TYPE OF BID.
1000C          CHNUMP(N)  - NUMBER OF BIDS OF CHARACTER TYPE N.
1010C
1020          COMMON /PCP/ NBIDS,CHNUMP(12)
1030          COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
1040          COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRBSK(350,12)
1050          INTEGER CHNUMP,BIDNUM,BIDTYP,CHRNAM,CHRBSK,PRCP
1051 STRING FILENAME
1052 111 PRINT,"ENTER FILENAME"
1053 INPUT,FILENAME
1054 OPEN(FILENAME,STATUS="OLD",UNIT=9,ERR=100)
1055 GOTO 110
1056 100 PRINT,"FILE NOT FOUND"
1057 GOTO 111
1060 110    IF(NCHAR.GT.1) GO TO 2
1070          NERR=0
```

```
      DO 1 NBIDS=1,350
      BIDNUM(NBIDS)=NBIDS
 1    READ(9,*,END=11) BIDPRI(NBIDS),BIDNEG(NBIDS)
      GO TO 10
    2 DO 200 I=1,NCHAR
 200  CHNUMP(I)=0
      DO 9 NBIDS=1,350
      BIDNUM(NBIDS)=NBIDS
 3    READ(9,*,END=11) BIDPRI(NBIDS),BIDNEG(NBIDS),BIDTYP(NBIDS)
      IF(1.LE.BIDTYP(NBIDS).AND.BIDTYP(NBIDS).LE.NCHAR) GO TO 9
      PROP=BIDNUM(NBIDS)+NERR
      WRITE("     ",4) PROP,BIDTYP(NBIDS)
 4    FORMAT(' ',/,' *** ERROR ENCOUNTERED ON PROPOSAL',I4,'.'/
     &'     THE TYPE OF PROPOSAL ',I3,' DOES NOT MATCH WITH ANY OF',
     &' THOSE SUBMITTED.')
      NERR=NERR+1
      WRITE("     ",5)
PRINT,""
 5    FORMAT(' DO YOU WISH TO OMIT THIS PROPOSAL FROM THE ANALYSIS',
     &' AND CONTINUE? (YES OR NO)')
      CALL ANSWER($3)
    9 CHNUMP(BIDTYP(NBIDS))=CHNUMP(BIDTYP(NBIDS))+1
 10   NBIDS=301
 11   NBIDS=NBIDS-1
      RETURN
      END
C
C
C
C
C
      SUBROUTINE PROGRM(*,*)
      INTEGER SIML,PRIC,ANSWR,STP
      DATA LIST/'LIST'/,SIML/'SIML'/,PRIC/'PRIC'/,STP/'STOP'/
    1 WRITE("     ",2)
PRINT,""
    2 FORMAT(//' ENTER "LIST" - TO LIST DATA',/,7X,'"SIML" - TO',
     &' RUN SIMULATION',/,7X,'"PRIC" - TO PERFORM PRICING',
     &/,7X,'"STOP" - TO END PROCESSING')
      READ("     ",3) ANSWR
    3 FORMAT(A4)
      IF(ANSWR.EQ.LIST) RETURN
      IF(ANSWR.EQ.SIML) RETURN1
      IF(ANSWR.EQ.PRIC) RETURN2
      IF(ANSWR.EQ.STP) STOP
      WRITE("     ",4) ANSWR
    4 FORMAT(' *** ERROR ***',A4,' IS NOT ONE OF ALLOWABLE COMMANDS.',
     &/,19X,'REENTER ONE OF FOLLOWING COMMANDS.')
      GO TO 1
      END
C
```

```
C
      SUBROUTINE LIST(*)
C  LISTS DATA.
      COMMON /POP/ NBIDS,CHNUMP(12)
      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRESK(350,12)
      INTEGER CHNUMP,BIDNUM,BIDTYP,CHRNAM,CHRESK
      WRITE("     ",1) NBIDS
    1 FORMAT(//,' LISTING',//,' NUMBER OF BIDS IN DATA SET = ',I4,/)
      IF(NCHAR.EQ.1) GO TO 9
      WRITE("     ",2)
    2 FORMAT(' NUMBER OF BIDS PER CHARACTER TYPE.',/)
      DO 3 I=1,NCHAR
    3 WRITE("     ",4) I,(CHRNAM(I,J),J=1,2),CHNUMP(I)
    4 FORMAT(5X,I2,'.',2X,2A4,4X,I3)
      WRITE("     ",5)
    5 FORMAT(/,' DATA')
      WRITE("     ",6)
    6 FORMAT(10X,'BID PRICE',3X,'NEGOTIATED PRICE',3X,'CODE',/)
      DO 7 I=1,NBIDS
    7 WRITE("     ",8) I,BIDPRI(I),BIDNEG(I),BIDTYP(I)
    8 FORMAT(4X,I3,'.',1X,F10.3,4X,F10.3,8X,I2)
      RETURN1
    9 WRITE("     ",5)
      WRITE("     ",10)
   10 FORMAT(10X,'BID PRICE',5X,'NEGOTIATED PRICE',/)
      DO 11 I=1,NBIDS
   11 WRITE("     ",12) I,BIDPRI(I),BIDNEG(I)
   12 FORMAT(5X,I3,'.',2X,F10.3,4X,F10.3)
      RETURN1
      END
C
C
      SUBROUTINE ANSWER(*)
      LOGICAL ARG
      INTEGER ANSWR,YES,NO
      DATA YES/'YE'/,NO/'NO'/
    1 READ("     ",2) ANSWR
    2 FORMAT(A2)
      IF(ANSWR.EQ.YES) RETURN1
      IF(ANSWR.EQ.NO) STOP
      WRITE("     ",3)
 PRINT,""
    3 FORMAT(' REENTER "YES OR NO"')
      GO TO 1
      ENTRY ANSWR1(ARG)
      READ("     ",2) ANSWR
      ARG=.FALSE.
      IF(ANSWR.EQ.YES) ARG=.TRUE.
      RETURN
      END
C
```

```
C
      SUBROUTINE SIMULA(*)
C  PERFORMS SIMULATION RUN.
C
      COMMON /RANGE/ NBASKL,NBASKH,NBASK
      COMMON /REPSEL/ NREP,RAND,IMBAL,NCODE
      LOGICAL RAND
      CALL CHORD(1)
      CALL SMINPU
      CALL SMINIT
      NPROMP=0
      DO 3 IREP=1,NREP
      NPROMP=NPROMP+1
      IF(NPROMP.LT.30) GO TO 100
      WRITE(3,200)
 200  FORMAT('ENTER "C" TO CONTINUE')
      READ(1,300) NPROPT
 300  FORMAT(A1)
      NPROMP=0
 100  IF(RAND) GO TO 1
      CALL CHOICE($2)
   1  CALL RANDOM
   2  DO 3 NBASK=NBASKL,NBASKH
      CALL BDORD
      CALL BASKT
      CALL SWAP
   3  CALL OVERAW
      CALL CHORD(3)
      CALL PCTMOM
      CALL RNSTAT
      DO 4 NBASK=NBASKL,NBASKH
4     CALL HISTO
      CALL RETABL (21,NREP)
      RETURN1
      END
C
```

```
C
      SUBROUTINE CHORD(K9)
      COMMON /POP/ NBIDS,CHNUMP(12)
      COMMON /BID/ BIDNUM(250),BIDPRT(250),BIDNEG(250),BIDTYP(250)
      COMMON /CHRCTP/ NCHAR,CHRNAM(12,2),CHRBSK(250,12)
      INTEGER BIDTYP,BIDNUM,KK(12),ORIG(250)
      REAL SUM(12)
      IF(K9.EQ.1)GOTO 772
      DO 639 J=1,NBIDS
639   BIDTYP(BIDNUM(J))=ORIG(BIDNUM(J))
      RETURN
772   DO 546 I=1,NBIDS
546   ORIG(BIDNUM(I))=BIDTYP(BIDNUM(I))
      DO 541 I=1,NCHAR
541   SUM(I)=0.0
      DO 779 K=1,NBIDS
779   SUM(BIDTYP(BIDNUM(K)))=SUM(BIDTYP(BIDNUM(K)))+BIDPRT(BIDNUM(K))
      K2=1
      DO 141 N=1,NCHAR
      BMAX=-10.**6
      DO 151 I=1,NCHAR
      IF(BMAX.GE.SUM(I))GOTO151
      KK(I)=K2
      BMAX=SUM(I)
      IM=I
151   CONTINUE
      SUM(IM)=-10.**6
141   K2=K2+1
      DO 557 M=1,NBIDS
557   BIDTYP(BIDNUM(M))=KK(BIDTYP(BIDNUM(M)))
      RETURN
      END
C
C
      SUBROUTINE SMINPU
      CALL BSKRAN
      CALL NUMSUB
      CALL TYPSEL
      CALL NUMREP
      RETURN
      END
C
C
      SUBROUTINE BSKRAN
C  INPUTS LOW AND HIGH RANGE OF BASKETS INTO - NBASKL,NBASKH.
      COMMON /RANGE/ NBASKL,NBASKH,NBASK
    1 WRITE("    ",2)
    2 FORMAT(' ENTER RANGE FOR NUMBER OF BASKETS.')
      READ("    ",*) NBASKL,NBASKH
      IF(2.LE.NBASKL.AND.NBASKL.LE.NBASKH.AND.NBASKH.LE.10) RETURN
      WRITE("    ",3)
    3 FORMAT(' *** ERROR *** RANGE OF BASKETS IS NOT BETWEEN ',
     &'2 AND 10 INCLUSIVE',/,15X,'OR LOW RANGE IS LARGER THAN HIGH',
     &' RANGE.',/,15X,'DO YOU WISH TO REENTER? (YES OR NO)')
      CALL ANSWER($1)
      STOP
      END
C
```

```
2580C
2590         SUBROUTINE NUMSUB
2600C  INPUTS NUMBER OF BIDS IN SUBPOPULATION INTO - NSUBP.
2610         COMMON /POP/ NBIDS,CHNUMP(12)
2620         COMMON /SUBPOP/ NSUBP,CHNUMS(12)
2630         COMMON /RANGE/ NBASKL,NBASKH,NBASK
2640         INTEGER CHNUMP,CHNUMS
2650      1  WRITE("     ",2)
2651 PRINT,""
2660      2  FORMAT(' ENTER SUBPOPULATION SIZE.')
2670         READ("     ",*) NSUBP
2680         IF(NSUBP.GE.NBASKH.AND.NSUBP.LE.NBIDS) RETURN
2690         WRITE("     ",3) NSUBP,NBASKH,NBIDS
2700      3  FORMAT(' *** ERROR *** VALUE OF SUBPOPULATION SIZE,',I5,
2710        &' IS NOT BETWEEN',I3,/,15X,' AND',I4,' INCLUSIVE.',/,15X,
2720        &'DO YOU WISH TO REENTER SUBPOPULATION SIZE? (YES OR NO)')
2730         CALL ANSWER($1)
2740         STOP
2750         END
2760C
2770C
2780         SUBROUTINE TYPSEL
2790C  INPUTS TYPE OF SUBPOPULATION SELECTION
2800         COMMON /REPSEL/ NREP,RAND,IMBAL,NCODE
2810         COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRESK(350,12)
2820         INTEGER CHRNAM,CHRESK
2830         LOGICAL RAND
2840         RAND=.TRUE.
2850         IF(NCHAR.EQ.1) RETURN
2860  7      WRITE("     ",1)
2861 PRINT,""
2870  1      FORMAT(' THERE ARE THREE SUBPOPULATION SELECTION OPTIONS:'
2880        &,/,10X,'1. SIMPLE RANDOM SAMPLING'
2890        &,/,10X,'2. STRATIFIED RANDOM SAMPLING - PROPORTIONAL ALLOCATION'
2900        &,/,10X,'3. STRATIFIED RANDOM SAMPLING - USER ALLOCATION'
2910        &,/,' WHICH OPTION DO YOU PREFER? (1,2 OR 3)')
2920         READ("     ",*) NCODE
2930         GO TO (4,5,6), NCODE
2940         WRITE("     ",3) NCODE
2950  3      FORMAT(' ','*** ERROR ***',I5,' IS NOT OPTION 1,2 OR 3.',/,
2960        &' DO YOU WISH TO REENTER? (YES OR NO)')
2970         CALL ANSWER($7)
2980         STOP
2990  4      RETURN
3000  5      RAND=.FALSE.
3010         CALL PROPSL
3020         RETURN
3030  6      RAND=.FALSE.
3040         CALL USERCH
3050         RETURN
3060         END
3070C
```

```
C
        SUBROUTINE NUMREP
C  INPUTS NUMBER OF REPLICATION INTO - NREP.
        COMMON /REPSEL/ NREP,RAND,IMBAL,NCODE
        LOGICAL RAND
     1  WRITE("     ",2)
 PRINT,""
     2  FORMAT(' ENTER NUMBER OF REPLICATIONS.')
        READ("     ",*) NREP
        IF(NREP.GE.1) RETURN
        WRITE("     ",3) NREP
     3  FORMAT(' *** ERROR *** NUMBER OF REPLICATIONS,',I4,'IS NOT'
       1' GREATER THAN 0.',/,15X,'DO YOU WISH TO REENTER NUMBER OF '
       2'REPLICATIONS? (YES OR NO)')
        CALL ANSWER(31)
        STOP
        END
C
C
        SUBROUTINE SMINIT
C  INITITALIZES FREQ(I,J)=0  I=NBASKL,NBASKH, J=1,21
                BELOW(I)=0   I=NBASKL,NBASKH
                OUTOFR(I)=0  I=NBASKL,NBASKH
        COMMON /RANGE/ NBASKL,NBASKH,NBASK
        COMMON /RANDM/ PERMUT(350),IX
        COMMON /HIST/ FREQ(21,10),BELOW(10),OUTOFR(10),OVRUNS(50,10)
        COMMON /ERROR/ PCTEPR,NERROR(10),ERRSUM(10),ERRSQ(10),ERRMOM(10,2)
        INTEGER FREQ,BELOW,BSK,OUTOFR,PERMUT
        DO 1 BSK=NBASKL,NBASKH
        BELOW(BSK)=0
        OUTOFR(BSK)=0
        ERRSUM(BSK)=0.
        ERRSQ(BSK)=0.
        NERROR(BSK)=0
        DO 1 J=1,21
     1  FREQ(J,BSK)=0
        ENTRY PRINIT
 IX=RRAND(0)
        RETURN
        END
C
C
```

```
C
C
      SUBROUTINE PROPSL
      COMMON /POP/ NBIDS,CHNUMP(12)
      COMMON /SUBPOP/ NSUBP,CHNUMS(12)
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRBSK(350,12)
      INTEGER CHNUMP,CHNUMS,CHRNAM,CHRBSK,ORDSZ(12)
      REAL XSMPST(12)
      NTOTSZ=0
      EPS=1.E-4
      XPROP=FLOAT(NSUBP)/FLOAT(NBIDS)
      DO 15 I=1,NCHAR
      XSMPST(I)=FLOAT(CHNUMP(I))*XPROP
      CHNUMS(I)=XSMPST(I)+ EPS
   15 NTOTSZ=NTOTSZ + CHNUMS(I)
      NDIFF=NSUBP-NTOTSZ
      IF(NDIFF.EQ.0) RETURN
      CALL ORDER(12,ORDSZ,XSMPST,NCHAR)
      KNT=1
      I=1
    7 IF(KNT.GT.NDIFF) RETURN
      IF(XSMPST(ORDSZ(I)).LE.0.0) GO TO 3
      CHNUMS(ORDSZ(I)) = CHNUMS(ORDSZ(I)) + 1
      KNT=KNT + 1
    3 I=I+1
      GO TO 7
      END
C
C
      SUBROUTINE USERCH
      COMMON /POP/ NBIDS,CHNUMP(12)
      COMMON /SUBPOP/ NSUBP,CHNUMS(12)
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRBSK(350,12)
      INTEGER CHNUMP,CHNUMS,CHRNAM,CHRBSK
      WRITE("     ",99)
   99 FORMAT(' ','STRATA COMPOSITION OF POPULATION',/)
      WRITE("     ",100) (CHRNAM(I,1),CHRNAM(I,2),CHNUMP(I),I=1,NCHAR)
  100 FORMAT(4(' ',A4,A4,1X,I3,2X))
    1 NPROP=0
      WRITE("     ",2) NCHAR,NSUBP
    2 FORMAT(/,' ENTER ',I2,' STRATA SAMPLE SIZES FOR SUBPOPULATION'
     &,/,' SUM OF THE SIZES MUST EQUAL ',I3,/)
      DO 12 I=1,NCHAR
      WRITE("     ",3) (CHRNAM(I,J),J=1,2)
    3 FORMAT(' ',2A4)
    4 CONTINUE
      READ("     ",*) CHNUMS(I)
      IF(0.LE.CHNUMS(I).AND.CHNUMS(I).LE.CHNUMP(I)) GO TO 11
      WRITE("     ",5) (CHRNAM(I,J),J=1,2),CHNUMP(I)
    5 FORMAT(' *** ERROR *** STRATA SAMPLE SIZE FOR CHARACTER TYPE, '
     &,2A4,' IS LESS THAN 0',/,5X,'OR GREATER THAN NUMBER OF BIDS',
     &' OF THAT TYPE,',I3,'.',/,15X,'DO YOU WISH TO REENTER',
     &' STRATA SAMPLE SIZE? (YES OR NO)')
      CALL ANSWER(*6)
    6 WRITE("     ",7)
  PRINT,""
 7    FORMAT(' REENTER VALUE.')
      GO TO 4
 11   NPROP=NPROP + CHNUMS(I)
 12   CONTINUE
      IF(NPROP.EQ.NSUBP) RETURN
      WRITE("     ",13) NPROP,NSUBP
 13   FORMAT(' *** ERROR *** SUM OF STRATA SAMPLE SIZES,',I4,
     &' DOES NOT EQUAL',/,15X,'SUBPOPULATION SIZE,',I4,'.',/,15X
     &'DO YOU WISH TO REENTER STRATA SAMPLE SIZES? (YES OR NO)')
      CALL ANSWER(*1)
      STOP
      END
C
C
      SUBROUTINE CHOICE(*)
C SELECTS NSUBP BIDS ACCORDING TO THE PROPORTION NUMBERS INPUTED.
C
      COMMON /POP/ NBIDS,CHNUMP(12)
      COMMON /SUBPOP/ NSUBP,CHNUMS(12)
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRBSK(350,12)
      INTEGER CHNUMP,CHNUMS,CHRNAM,CHRBSK
      INTEGER CHARA
      CALL CHBASK(NBIDS)
      DO 1 CHARA=1,NCHAR
      CALL PERM(CHNUMS(CHARA),CHNUMP(CHARA))
    1 CALL SWITCH(CHRBSK(1,CHARA),CHNUMP(CHARA))
      CALL BSKORD
      RETURN1
      END
C
C
      SUBROUTINE RANDOM
C SELECTS THE SUBPOPULATION FROM THE PARENT POPULATION RANDOMLY.
C
      COMMON /POP/ NBIDS,CHNUMP(12)
      COMMON /SUBPOP/ NSUBP,CHNUMS(12)
      COMMON /RANDM/ PERMUT(350),IX
      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
      INTEGER CHNUMP,CHNUMS,BIDNUM,BIDTYP,PERMUT
      CALL PERM(NSUBP,NBIDS)
      CALL SWITCH(BIDNUM(1),NBIDS)
      CALL CHBACK(NSUBP)
      CALL BSKORD
      RETURN
      END
C
```

```
4500C
4510          SUBROUTINE PERM(STOP1,END)
4520          COMMON /RANDM/ PERMUT(350),IX
4530          INTEGER PERMUT,COUNT,STOP1,END,END1
4540          COUNT=0
4550          DO 1 I=1,END
4560        1 PERMUT(I)=I
4570          END1=END-1
4580          DO 2 I=1,END1
4590          J=END-I+1
4600          K=IRAND(J)+I-1
4610          L=PERMUT(I)
4620          PERMUT(I)=PERMUT(K)
4630          PERMUT(K)=L
4640          COUNT=COUNT+1
4650          IF(COUNT.EQ.STOP1) RETURN
4660        2 CONTINUE
4670          RETURN
4680          END
4690C
4700C
4710          FUNCTION IRAND(J)
4720          COMMON /RANDM/ PERMUT(350),IX
4730          INTEGER PERMUT
4750 RAND=RND(IX)
4760          IRAND=RAND*J+1
4770          RETURN
4780          END
4790C
4800C
4810          SUBROUTINE SWITCH(ARRAY,SIZE)
4820          INTEGER ARRAY(350),PERMUT,ARRAY1(350),SIZE
4830          COMMON /RANDM/ PERMUT(350),IX
4840          DO 1 I=1,SIZE
4850        1 ARRAY1(I)=ARRAY(PERMUT(I))
4860          DO 2 I=1,SIZE
4870        2 ARRAY(I)=ARRAY1(I)
4880          RETURN
4890          END
4900C
4910C
```

```
4920C
4930C
4940        SUBROUTINE CHBASK(BIDS)
4950C   PUTS BIDS INTO BASKETS - CHRBSK(N,K) - BY TYPE.
4960C
4970        COMMON /SUBPOP/ NSUBP,CHNUMS(12)
4980        COMMON /RANGE/ NBASKL,NBASKH,NBASK
4990        COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
5000        COMMON /BASK/ BASKET(10,350),BASK1(10,160),BTOT(10),BNEG(10)
5010        COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRBSK(350,12)
5020        COMMON /REPSEL/ NREP,RAND,IMBAL,NCODE
5030        INTEGER CHNUMS,BIDNUM,BIDTYP,BASKET,BASK1,CHRNAM,CHRBSK
5040        INTEGER CHARA,CHRNUM(12),BIDS
5050        LOGICAL RAND
5060        IMBAL=0
5070        IF(NCHAR.EQ.1) GO TO 9
5080        DO 1 CHARA=1,NCHAR
5090    1   CHRNUM(CHARA)=0
5100        DO 2 NBID=1,BIDS
5110        NUMBID=BIDNUM(NBID)
5120        INDEX=BIDTYP(NUMBID)
5130        CHRNUM(INDEX)=CHRNUM(INDEX)+1
5140    2   CHRBSK(CHRNUM(INDEX),INDEX)=NUMBID
5150        IF(.NOT.RAND) RETURN
5160        DO 8 CHARA=1,NCHAR
5170        IF(CHRNUM(CHARA).GE.NBASKH) GO TO 8
5180        IMBAL=IMBAL+1
5190    8   CHNUMS(CHARA)=CHRNUM(CHARA)
5200        RETURN
5210    9   CHNUMS(1)=NSUBP
5220        DO 10 NBID=1,BIDS
5230    10  CHRBSK(NBID,1)=BIDNUM(NBID)
5240        RETURN
5250        END
5260C
```

```
C
      SUBROUTINE BSKORD
C  PUTS BIDS WITHIN EACH CHARACTER TYPE BASKET IN ORDER
C  FROM LARGEST TO SMALLEST.
C
      COMMON /SUBPOP/ NSUBP,CHNUMS(12)
      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRBSK(350,12)
      INTEGER CHNUMS,BIDNUM,BIDTYP,CHRNAM,CHRBSK
      INTEGER CHARA,FIRST,RECENT,TEMP
      DO 5 CHARA=1,NCHAR
      RECENT=1
      LAST=CHNUMS(CHARA)
      IF(LAST.LE.1) GO TO 5
    1 FIRST=RECENT+1
      DO 2 J=FIRST,LAST
      JM1=J-1
      IF(BIDPRI(CHRESK(JM1,CHARA)).GE.BIDPRI(CHRESK(J,CHARA))) GO TO 2
      RECENT=JM1
      TEMP=CHRBSK(JM1,CHARA)
      CHRBSK(JM1,CHARA)=CHRESK(J,CHARA)
      CHRBSK(J,CHARA)=TEMP
    2 CONTINUE
      IF(RECENT+1.EQ.FIRST) GO TO 5
      LAST=RECENT
      J=LAST
    3 JM1=J-1
      IF(BIDPRI(CHRESK(JM1,CHARA)).GE.BIDPRI(CHRBSK(J,CHARA))) GO TO 4
      RECENT=J
      TEMP=CHRBSK(JM1,CHARA)
      CHRBSK(JM1,CHARA)=CHRESK(J,CHARA)
      CHRECK(J,CHARA)=TEMP
    4 J=J-1
      IF(J.GE.FIRST) GO TO 3
      IF(RECENT.LT.LAST) GO TO 1
    5 CONTINUE
      RETURN
      END
C
```

```
C
C
C
      SUBROUTINE BDORD
C DETERMINES ORDER IN WHICH BIDS WILL BE PLACED IN GROUPS OF NBASK.
C CREATES - BIDOR(N) - WHICH IS THE INDEX OF THE CHARACTERISTIC
C OF THE NTH BID GROUPING.
C       NGROUP - NUMBER OF SUCH GROUPINGC.
C
      COMMON /SUBPOP/ NSUBP,CHNUMS(12)
      COMMON /RANGE/ NBASKL,NBACKH,NBASK
      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
      COMMON /BIDORD/ BIDOR(100),NGROUP
      COMMON /BASK/ BASKET(10,350),BASK1(10,100),BTOT(10),BNEG(10)
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRBSK(350,12)
      COMMON /AVGORD/AVG(12),NCHR(12),NUMBID,NCHAR1,MAX
      INTEGER CHNUMS,BIDNUM,BIDTYP,BIDOR,BASKET,BASK1,CHRNAM,CHRBSK
      INTEGER CHARA
      NCHAR1=NCHAR
      NGROUP=0
      NUMBID=0
      IF(NCHAR.EQ.1) GO TO 4
      DO 1 CHARA=1,NCHAR
    1 NCHR(CHARA)=0
      DO 3 N=1,400
      IF(NUMBID.GE.NSUBP) RETURN
      CALL MAXAVG
      NGROUP=NGROUP+1
      BIDOR(NGROUP)=MAX
    3 CONTINUE
      RETURN
    4 NGROUP=NSUBP/NBACK
      IF(NBASK*NGROUP.NE.NSUBP) NGROUP=NGROUP+1
      DO 5 NGROU=1,NGROUP
    5 BIDOR(NGROU)=1
      NCHR(1)=NSUBP
      RETURN
      END
C
```

```
C
      SUBROUTINE MAXAVG
C   CALCULATES AVERAGE OF THE NEXT NBASK BIDS OF EACH CHARACTERISTI
C   TYPE, AND RETURNS THE INDEX OF THE CHARACTERISTIC TYPE HAVING T
C   LARGEST AVERAGE.
C        NCHR(N) - INDEX OF POSITION TO BEGIN AVERAGE OF BIDS FOR
C                  CHARACTERISTIC TYPE N.
C
      COMMON /SURPOP/ NSURP,CHNUMS(12)
      COMMON /RANGE/ NBASKL,NBASKH,NBASK
      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRBSK(350,12)
      COMMON /AVGORD/AVG(12),NCHR(12),NUMBID,NCHAR1,MAX
      INTEGER CHNUMS,BIDNUM,BIDTYP,CHRNAM,CHRBSK
      INTEGER START,END
      MAX=1
      DO 2 N=1,NCHAR1
      AVG(N)=0
      START=NCHR(N)+1
      END=NCHR(N)+NBASK
      IF(END.GT.CHNUMS(N)) END=CHNUMS(N)
      IF(START.GT.END) GO TO 2
      DO 1 J=START,END
 1    AVG(N)=AVG(N)+BIDPRI(CHRBSK(J,N))
      AVG(N)=AVG(N)/(END-START+1)
 3    IF(AVG(N).LT.AVG(MAX)) GO TO 2
      MAX=N
      MAXBDS=END-START+1
 2    CONTINUE
      NUMBID=NUMBID+MAXBDS
      NCHR(MAX)=NCHR(MAX)+MAXBDS
      RETURN
      END
C
C
```

```
C
C
      SUBROUTINE BASKT
C   PLACES BIDS INTO BASKETS ACCORDING TO BIDOR AND BASKET TOTALS.
C
      COMMON /RANGE/ NBASKL,NBASKH,NBASK
      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
      COMMON /BASK/ BASKET(10,350),BASK1(10,160),BTOT(10),BNEG(10)
      COMMON /BIDORD/ BIDOR(160),NGROUP
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRBSK(350,12)
      COMMON /AVGORD/AVG(12),NCHR(12),NUMBID,NCHAR1,MAX
      INTEGER BIDNUM,BIDTYP,BASKET,BASK1,BIDOR,CHRNAM,CHRBSK
      INTEGER START(12),UP1(10),ORJ,ORDER1(10),ORDER2(10),BIDORI
      DO 1 N=1,NBASK
      BTOT(N)=0
 1    UP1(N)=0
      DO 20 N=1,NCHAR
20    START(N)=0
      DO 6 I=1,NGROUP
      BIDORI=BIDOR(I)
      NUMLEF=NCHR(BIDORI)-START(BIDORI)
      CALL ORDER(10,ORDER1,BTOT,NBASK)
      IF(NUMLEF.LT.NBASK) GO TO 3
      DO 2 J=1,NBASK
      ORJ=ORDER1(J)
      INDEX=CHRBSK(START(BIDORI)+J,BIDORI)
      PRICE=BIDPRI(INDEX)
      BASKET(ORJ,I)=INDEX
 2    BTOT(ORJ)=BTOT(ORJ)+PRICE
      START(BIDORI)=START(BIDORI)+NBASK
      GO TO 6
 3    CALL UPONE(ORDER1,ORDER2,NBASK,UP1,NUMLEF)
      DO 4 J=1,NUMLEF
      ORJ=ORDER2(J)
      INDEX=CHRBSK(START(BIDORI)+J,BIDORI)
      PRICE=BIDPRI(INDEX)
      BASKET(ORJ,I)=INDEX
 4    BTOT(ORJ)=BTOT(ORJ)+PRICE
      NMLEF1=NUMLEF+1
      DO 5 J=NMLEF1,NBASK
      ORJ=ORDER2(J)
C   WHEN NUMBER OF BIDS LEFT OF ANY PARTICULAR CHARACTERISTIC TYPE
C   IS LESS THAN NUMBER OF BASKETS, THE BIDS ARE PLACED INTO
C   BASKETS BY THE USUAL METHOD AND THE UNFILLED BASKETS RECEIVE
C   THE VALUE -1 INTO BASKET(N,K), THE INDICE ARRAY.
C                       FOR LATER CORRECTION OF BTOT(N).
 5    BASKET(ORJ,I)=-1
 6    CONTINUE
      RETURN
      END
C
```

```
7040C
7050          SUBROUTINE ORDER(M,ORDER1,BTOT,NBASK)
7060C   ORDERS THE BASKET INDICES INTO - ORDER1 - FROM SMALLEST BASKET
7070C   TO LARGEST BASKET.
7080C
7090          INTEGER ORDER1(M)
7100          REAL BTOT(M)
7110          DO 1 N=1,NBASK
7120    1     ORDER1(N)=N
7130          NBASK1=NBASK-1
7140          DO 3 N=1,NBASK1
7150          MIN=N
7160          NBASK2=N+1
7170          DO 2 K=NBASK2,NBASK
7180          IF(BTOT(ORDER1(MIN)).LE.BTOT(ORDER1(K))) GO TO 2
7190          ITEMP=ORDER1(MIN)
7200          ORDER1(MIN)=ORDER1(K)
7210          ORDER1(K)=ITEMP
7220    2     CONTINUE
7230    3     CONTINUE
7240          RETURN
7250          END
7260C
7270C
7280          SUBROUTINE UPONE(ORDER1,ORDER2,NBASK,UP1,NUMLEF)
7290C   DETERMINES WHICH BASKETS HAVE ONE MORE BID THAN OTHERS AND RETURNS
7300C   - ORDER2 - THE NEW ORDER OF BASKET INDICES REFLECTING SMALLEST TO
7310C   LARGEST AND NUMBER OF BIDS IN nASKETS.
7320C
7330          INTEGERPORDER1(10),ORDER2(10),UP1(10),ENDJ,STARTJ
7340          STARTJ=0
7350          DO 1 I=1,NBASK
7360          IF(UP1(ORDER1(I)).EQ.1) GO TO 1
7370          STARTJ=STARTJ+1
7380          ORDER2(STARTJ)=ORDER1(I)
7390    1     CONTINUE
7400          IF(STARTJ.NE.0) GO TO 3
7410          DO 2 J=1,NBASK
7420    2     ORDER2(J)=ORDER1(J)
7430          GO TO 5
7440    3     IF(STARTJ.EQ.NBASK) GO TO 8
7450          ENDJ=STARTJ
7460          DO 4 I=1,NBASK
7470          IF(UP1(ORDER1(I)).EQ.0) GO TO 4
7480          ENDJ=ENDJ+1
7490          ORDER2(ENDJ)=ORDER1(I)
7500    4     CONTINUE
7510    5     IF(STARTJ.GE.NUMLEF) GO TO 10
7520          DO 6 J=1,NUMLEF
7530          IF(UP1(ORDER2(J)).EQ.0) UP1(ORDER2(J))=0
7540    6     IF(UP1(ORDER2(J)).EQ.1) UP1(ORDER2(J))=1
7550          NMLEF1=NUMLEF+1
7560          DO 7 J=NMLEF1,NBASK
7570    7     UP1(ORDER2(J))=0
7580          RETURN
7590    8     NMLEF1=NUMLEF+1
7600          DO 9 J=NMLEF1,NBASK
7610    9     UP1(ORDER2(J))=0
7620    10    DO 11 J=1,NUMLEF
7630    11    UP1(ORDER2(J))=1
7640          RETURN
7650          END
7660C
```

```
C
      SUBROUTINE SWAP
C  PERFORMS A BID SWAP BETWEEN BASKETS TO BALANCE BASKET TOTALS.
C     LGBSK - INDEX OF LARGEST BASKET WITH RESPECT TO BASKET TOTALS.
C     SMBSK - INDEX OF SMALL BASKDFT.
C
      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
      COMMON /BASK/ BASKET(10,350),BASK1(10,100),BTOT(10),BNEG(10)
      COMMON /RANGE/ NBASKL,NBASKH,NBASK
      COMMON /SWAP1/LGBSK,LGIND,SMBSK,SMIND,STARTC(12),CHAR
      INTEGER BIDNUM,BIDTYP,BASKET,BASK1
      INTEGER LGBSK,LGIND,SMBSK,SMIND,STARTC,CHAR
      INTEGER ORDER1(10),BSK,SWTCH1,SWTCH2
      CALL SETUP
 1    CALL ORDER(10,ORDER1,BTOT,NBASK)
      DO 3 N=2,NBASK
      LGBSK=ORDER1(NBACK-N+2)
      NBASK1=NBASK-1
      DO 2 K=1,NBASK1
      SMBSK=ORDER1(K)
      DIFF=BTOT(LGBSK)-BTOT(SMBSK)
      IF(DIFF.EQ.0.) GO TO 3
      CALL DIFFNT(SWTCH2,DIFF)
      IF(SMIND.NE.0) GO TO 4
 2    CONTINUE
 3    CONTINUE
      RETURN
 4    DIFF=BIDPRI(BASK1(LGBSK,LGIND))-BIDPRI(BASK1(SMBSK,SMIND))
      BTOT(LGBSK)=BTOT(LGBSK)-DIFF
      BTOT(SMBSK)=BTOT(SMBSK)+DIFF
      CALL PLACE
      GO TO 1
      END
C
C
      SUBROUTINE SETUP
C  INITIALIZES - BASK1(N,K) - TO BE THE SAME AS - BASKET(N,K) - BUT
C  WITH BIDS PLACED TOGETHER WITHIN BASKETS BY CHARACTERISTIC TYPE.
C     NOTE: BIDS WILL ALSO BE IN ORDER FROM LARGEST TO SMALLEST
C           WITHIN CHARACTER GROUPINGS SINCE BASKET WAS ORDERED.
C
      COMMON /RANGE/ NBASKL,NBASKH,NBASK
      COMMON /BIDORD/ BIDOR(100),NGROUP
      COMMON /BASK/ BASKET(10,350),BASK1(10,100),BTOT(10),BNEG(10)
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRBSK(350,12)
      COMMON /SWAP1/LGBSK,LGIND,SMBSK,SMIND,STARTC(12),CHAR
      INTEGER BIDOR,BASKET,BASK1,CHRNAM,CHRBSK
      INTEGER LGBSK,LGIND,SMBSK,SMIND,STARTC,CHAR
      INTEGER COUNT(12),POS,BSK
      DO 1 N=1,NCHAR
 1    COUNT(N)=0
      DO 2 N=1,NGROUP
      NUMB=BIDOR(N)
 2    COUNT(NUMB)=COUNT(NUMB)+1
      STARTC(1)=1
      DO 3 N=2,NCHAR
      STARTC(N)=COUNT(N-1)+STARTC(N-1)
 3    COUNT(N-1)=0
      COUNT(NCHAR)=0
      DO 5 N=1,NGROUP
      NUMB=BIDOR(N)
      POS=STARTC(NUMB)+COUNT(NUMB)
      DO 4 BSK=1,NBASK
 4    BASK1(BSK,POS)=BASKET(BSK,N)
 5    COUNT(NUMB)=COUNT(NUMB)+1
      RETURN
      END
C
C
      SUBROUTINE DIFENT(DIFF,DIFF1)
C   DETERMINES IF/WHICH BID SWAPS FOR GIVEN BASKETS WILL PRODUCE A
C   BETTER BALANCE.
C       LGIND - INDEX OF BID TO BE SWAPPED OUT OF LARGE BASKET.
C       SMIND - INDEX OF BID TO BE SWAPPED OUT OF SMALL BASKET.
C
      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
      COMMON /BIDORD/ BIDOR(160),NGROUP
      COMMON /BASK/ BASKET(10,350),BASK1(10,160),BTOT(10),BNEG(10)
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRBSK(350,12)
      COMMON /SWAP1/LGBSK,LGIND,SMBSK,SMIND,STARTC(12),CHAR
      INTEGER BIDNUM,BIDTYP,BIDOR,BASKET,BASK1,CHRNAM,CHRBSK
      INTEGER LGBSK,LGIND,SMBSK,SMIND,STARTC,CHAR
      INTEGER SWTCH1,SWTCH2,STRT,STP,CHARA,EQUAL
      EQUAL=0
      SMIND=0
      DIFF1=0
      DIFF12=DIFF/2.
      DIFF1D=ABS(DIFF1-DIFF12)
      DO 3 CHARA=1,NCHAR
      STRT=STARTC(CHARA)
      STP=NGROUP
      IF(CHARA.LT.NCHAR)STP=STARTC(CHARA+1)-1
      DO 2 SWTCH1=STRT,STP
      IND1=BASK1(LGBSK,SWTCH1)
      IF(IND1.EQ.-1) GO TO 2
      PRICE1=BIDPRI(IND1)
      DO 1 SWTCH2=STRT,STP
      IND2=BASK1(SMBSK,SWTCH2)
      IF(IND2.NE.-1.) GO TO 10
      DIFF2=PRICE1
      GO TO 12
 10   PRICE2=BIDPRI(IND2)
      DIFF2=PRICE1-PRICE2
 12   IF(DIFF2.LE.0.) GO TO 1
      IF(DIFF2.GE.DIFF) GO TO 2
      DIFF2D=ABS(DIFF2-DIFF12)
      IF(DIFF2D.LT.0) GO TO 14
      IF(DIFF2D.GT.DIFF1D) GO TO 2
      GO TO 16
 14   IF(DIFF2D.GT.DIFF1D) GO TO 1
 16   IF(DIFF2D.EQ.DIFF1D) GO TO 20
      EQUAL=0
      DIFF1=DIFF2
      LGIND=SWTCH1
      SMIND=SWTCH2
      CHAR=CHARA
C     IF(DIFF2.EQ.DIFF/2.) RETURN
      GO TO 1
20    IF(SMIND.EQ.0) GO TO 1
      EQUAL=EQUAL+1
1     CONTINUE
2     CONTINUE
3     CONTINUE
      IF(EQUAL.EQ.0) RETURN
      WRITE("    ",30) EQUAL
30    FORMAT(' ',I6,'  OTHER EQUAL SWAPS.')
      RETURN
      END
C
C
      SUBROUTINE PLACE
C   SWAPS BIDS.
C
      COMMON /BASK/ BASKET(10,350),BASK1(10,160),BTOT(10),BNEG(1
      COMMON /SWAP1/LGBSK,LGIND,SMBSK,SMIND,STARTC(12),CHAR
      INTEGER BASKET,BASK1
      INTEGER LGBSK,LGIND,SMBSK,SMIND,STARTC,CHAR
      TEMP=BASK1(LGBSK,LGIND)
      BASK1(LGBSK,LGIND)=BASK1(SMBSK,SMIND)
      BASK1(SMBSK,SMIND)=TEMP
      CALL PUT(LGBSK,LGIND,CHAR)
      CALL PUT(SMBSK,SMIND,CHAR)
      RETURN
      END
C
```

```
C
      SUBROUTINE PUT(BSK,IND,CHAR1)
C    MOVES 'NEW BID' INTO PROPER ORDERED POSITION WITHIN CHARACTER TY
C    GROUPING WITHIN BASKET.
C
      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
      COMMON /BASK/ BASKET(10,350),BASK1(10,160),BTOT(10),BNEG(10)
      COMMON /BIDORD/ BIDOR(160),NGROUP
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRBSK(350,12)
      COMMON /SWAP1/LGBSK,LGIND,SMBSK,SMIND,STARTC(12),CHAR
      INTEGER BIDNUM,BIDTYP,BASKET,BASK1,BIDOR,CHRNAM,CHRBSK
      INTEGER LGBSK,LGIND,SMBSK,SMIND,STARTC,CHAR
      INTEGER BSK,CHAR1,STRT,STP
      IF(BASK1(BSK,IND).EQ.-1) GO TO 4
      IF(IND.EQ.STARTC(CHAR1)) GO TO 2
      IF(BIDPRI(BASK1(BSK,IND)).LE.BIDPRI(BASK1(BSK,IND-1))) GO TO 2
      STRT=STARTC(CHAR1)
      STP=IND-1
      DO 1 K=STRT,STP
      KK=STP+STRT-K+1
      KK1=KK-1
      IF(BIDPRI(BASK1(BSK,KK)).LE.BIDPRI(BASK1(BSK,KK1))) RETURN
      TEMP=BASK1(BSK,KK)
      BASK1(BSK,KK)=BASK1(BSK,KK1)
      BASK1(BSK,KK1)=TEMP
    1 CONTINUE
      RETURN
    2 STRT=IND
      IF(STRT.EQ.NGROUP) RETURN
      STP=NGROUP-1
      IF(CHAR1.LT.NCHAR) STP=STARTC(CHAR1+1)-2
      IF(STRT.GT.STP) RETURN
      DO 3 K=STRT,STP
      IF(BASK1(BSK,K+1).EQ.-1) RETURN
      IF(BIDPRI(BASK1(BSK,K)).GE.BIDPRI(BASK1(BSK,K+1))) RETURN
      TEMP=BASK1(BSK,K)
      BASK1(BSK,K)=BASK1(BSK,K+1)
      BASK1(BSK,K+1)=TEMP
    3 CONTINUE
      RETURN
    4 STP=NGROUP
      IF(CHAR1.LT.NCHAR) STP=STARTC(CHAR1+1)-1
      IF(IND.EQ.STP) RETURN
      STP=STP-1
      DO 5 KK=LGIND,STP
    5 BASK1(BSK,KK)=BASK1(BSK,KK+1)
      BASK1(BSK,STP+1)=-1
      RETURN
      END
C
```

```
C
      SUBROUTINE OVERAW
C CALCULATES PERCENT OVER AWARD.
C
C
      COMMON /RANGE/ NBASKL,NBASKH,NBASK
      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
      COMMON /BASK/ BASKET(10,350),BASK1(10,160),BTOT(10),BNEG(10)
      COMMON /BIDORD/ BIDOR(160),NGROUP
      COMMON /ERROR/ PCTERR,NERROR(10),ERRSUM(10),ERRSQ(10),ERRMOM(10,2)
      INTEGER BIDNUM,BIDTYP,BASKET,BASK1,BIDOR
      INTEGER BSK
      BSK=IRAND(NBASK)
      BSKBID=0.
      BSKNEG=0.
      DO 1 I=1,NGROUP
      INDEX=BASK1(BSK,I)
      IF(INDEX.EQ.-1) GO TO 1
      BSKBID=BSKBID+BIDPRI(INDEX)
      BSKNEG=BSKNEG+BIDNEG(INDEX)
    1 CONTINUE
      SUMBID=0.
      TOTNEG=BSKNEG
      DECREM=BSKNEG/BSKBID
      DO 3 J=1,NBASK
      IF(J.EQ.BSK) GO TO 3
      DO 2 I=1,NGROUP
      INDEX=BASK1(J,I)
      IF(INDEX.EQ.-1) GO TO 2
      SUMBID=SUMBID+BIDPRI(INDEX)
      TOTNEG=TOTNEG+BIDNEG(INDEX)
    2 CONTINUE
    3 CONTINUE
      AWARD=BSKNEG+DECREM*SUMBID
      PCTERR=100.*(AWARD-TOTNEG)/TOTNEG
      ERRSUM(NBASK)=ERRSUM(NBASK)+PCTERR
      ERRSQ(NBASK)=ERRSQ(NBASK)+PCTERR**2
      NERROR(NBASK)=NERROR(NBASK)+1
      CALL FRECCY
      RETURN
      END
C
C
      SUBROUTINE FRECCY
C SETS UP THE FREQ ARRAY.
C
      COMMON /RANGE/ NBASKL,NBASKH,NBASK
      COMMON /HIST/ FREQ(21,10),BELOW(10),CUTOFR(10),OVRUNS(50,10)
      COMMON /ERROR/ PCTERR,NERROR(10),ERRSUM(10),ERRSQ.10),ERRMOM(10,2)
      INTEGER FREQ,BELOW,CUTOFR
      INDEX=PCTERR+11.5
      IF(INDEX.GT.21.OR.INDEX.LT.1) GO TO 1
      FREQ(INDEX,NBASK)=FREQ(INDEX,NBASK)+1
      RETURN
    1 CUTOFR(NBASK)=CUTOFR(NBASK)+1
      IF(INDEX.LT.1) BELOW(NBASK)=BELOW(NBASK)+1
      IF(CUTOFR(NBASK).LE.50) OVRUNS(CUTOFR(NBASK),NBASK)=PCTERR
      RETURN
      END
C
```

```
C
            SUBROUTINE PCTMOM
C CALCULATES MEAN AND STANDARD DEVIATION FOR PERCENT ERROR.
C                 ERRMOM(NBASK,1) - MEAN FOR BASKET NBASK.
C                 ERRMOM(NBASK,2) - STANDARD DEVIATION FOR BASKET NBASK.
C
            COMMON /RANGE/ NBASKL,NBASKH,NBASK
            COMMON /ERROR/ PCTERR,NERROR(10),ERRSUM(10),ERRSQ(10),ERRMOM(10,2)
            DO 1 NBASK=NBASKL,NBASKH
            ERRMOM(NBASK,1)=ERRSUM(NBASK)/NERROR(NBASK)
          1 ERRMOM(NBASK,2)=SQRT((ERRSQ(NBASK)-ERRSUM(NBASK)**2/NERROR(NBASK))
           &/(NERROR(NBASK)-1))
            RETURN
            END
C
C
            SUBROUTINE RNSTAT
C OUTPUTS RUN STATISTICS.
C
            COMMON /POP/ NBIDS,CHNUMP(12)
            COMMON /SUBPOP/ NSUBP,CHNUMS(12)
            COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRBSK(350,12)
            COMMON /REPSEL/ NREP,RAND,IMBAL,NCODE
            INTEGER CHNUMP,CHNUMS,CHRNAM,CHRBSK
            LOGICAL RAND
            WRITE("     ",1) NBIDS
        1   FORMAT(///,' RUN STATISTICS',//,5X,'POPULATION SIZE = ',I3,/)
            IF(NCHAR.EQ.1) GO TO 3
            WRITE("     ",2) ((CHRNAM(I,J),J=1,2),CHNUMP(I),I=1,NCHAR)
       2    FORMAT(10X,2A4,' = ',I3)
       3    WRITE("     ",4) NSUBP
       4    FORMAT(' ',//,5X,'SUBPOPULATION SIZE = ',I3,/)
            GO TO (6,11,12), NCODE
       6    WRITE("     ",7)
       7    FORMAT(/,5X,'SELECTION METHOD:  SIMPLE RANDOM SAMPLING.',/)
            WRITE("     ",8) IMBAL
       8    FORMAT(5X,'NUMBER OF SELECTIONS CAPABLE OF CAUSING IMBALANCE = '
           &,I3)
            GO TO 9
      11    WRITE("     ",5)
       5    FORMAT(/,5X,'SELECTION METHOD:  STRATIFIED RANDOM SAMPLING-',/,
           +27X,'PROPORTIONAL ALLOCATION',/)
            GO TO 14
      12    WRITE("     ",13)
      13    FORMAT(/,5X,'SELECTION METHOD:  STRATIFIED RANDOM SAMPLING-',/,
           &27X,'USER ALLOCATION',/)
      14    WRITE("     ",2) ((CHRNAM(I,J),J=1,2),CHNUMS(I),I=1,NCHAR)
       9    WRITE("     ",10) NREP
      10    FORMAT(/,5X,'NUMBER OF REPLICATIONS = ',I4,//)
            RETURN
            END
C
```

```
C
      SUBROUTINE HISTO
C
      COMMON /RANGE/ NBASKL,NBASKH,NBASK
      COMMON /REPSEL/ NREP,RAND,IMBAL
      COMMON /HIST/ FREQ(21,10),BELOW(10),OUTOFR(10),OVRUNS(50,10)
      COMMON /ERROR/ PCTERR,NERROR(10),ERRSUM(10),ERRSQ(10),ERRMOM(10,2
      INTEGER FREQ,OUTOFR,BELOW
      LOGICAL RAND
      WRITE("   ",1) NBASK
    1 FORMAT('-'/'-',31X,'SIMULATION RESULTS:'/22X,
     &'RELATIVE FREQUENCY OF PERCENT OVERAWARD'/34X,'WITH',
     &I3,' BASKETS')
      CALL HSTOGM(NREP,21,-10.,1.,FREQ(1,NBASK),BELOW(NBASK),.FALSE.)
      K=OUTOFR(NBASK)
      IF (K.EQ.0) GO TO 4
      WRITE("    ",2) K
    2 FORMAT('0',I3,' OVERAWARDS EXCEEDED THE RANGE OF THE CHART.')
      IF (K.GT.50) K=50
      WRITE("    ",3) (OVRUNS(I,NBASK),I=1,K)
    3 FORMAT(10F8.2)
    4 WRITE("    ",5) ERRMOM(NBASK,1),ERRMOM(NBASK,2)
    5 FORMAT(3X,'PERCENT ERROR:',3X,'MEAN =',F9.3,3X,'STD DEV =',F8.3)
      WRITE("    ",6)
    6 FORMAT('-'/'-')
      RETURN
      END
C
      SUBROUTINE HSTOGM(NTRIAL,NFREQ,CICTR,CIDEL,FREQS,BELOW,NORMLZ)
C
C THIS SUBROUTINE ACCEPTS AS INPUT AN INITIAL CLASS INTERVAL, A CLASS
C INTERVAL DELTA, THE NUMBER OF CLASS INTERVALS FOR WHICH FREQUENCY
C DATA WAS COLLECTED, THE TOTAL OF ALL FREQUENCIES IN THE EXPERIMEN
C AND AN ARRAY OF FREQUENCIES, EACH CORRESPONDING TO A CLASS INTERVAL.
C AS OUTPUT, IT PRODUCES A HISTOGRAM AND A LISTING OF THE RELATIVE
C AND CUMULATIVE RELATIVE FREQUENCIES FOR EACH CLASS INTERVAL.
C
C NTRIAL - TOTAL OF ALL FREQUENCIES.
C NFREQ - NUMBER OF CLASS INTERVALS.
C CICTR - INITIAL CLASS INTERVAL CENTER.
C CIDEL - WIDTH OF CLASS INTERVALS.
C FREQS(I) - FREQUENCY CORRESPONDING TO THE ITH CLASS INTERVAL, WHERE
C             'CICTR' IS THE FIRST CLASS INTERVAL.
C BELOW - THE NUMBER OF SAMPLES WHICH FELL PRIOR TO THE 1ST CLASS INTE
C
      INTEGER FREQS(NFREQ),BRCHRT(50),BLANK/' '/,PLUS/'X'/,ASTSK/'+'/
     &        ,BELOW
      REAL RFREQS(50)
      LOGICAL NORMLZ
C
```

```
C PRINT HEADER FOR HISTOGRAM.
        IF (NORMLZ) WRITE("    ",1)
     1 FORMAT('0  CLASS',6X,'CUM'/
      &        '  INTERVAL',5X,'REL',6X,'REL',4X,'BAR CHART'/
      &        '  CENTER',2(5X,'FREQ')/
      &       1X,29('-'),'I',50('-'))
        IF (.NOT.NORMLZ) WRITE("    ",5)
     5 FORMAT('0  CLASS',6X,'CUM',13X,'BAR CHART'/
      &        '  INTERVAL',5X,'REL',6X,'REL'/
      &        '  CENTER',2(5X,'FREQ'),7X,' 0.1  0.2  0.3  0.4  0.5 '
      &       ,' 0.6  0.7  0.8  0.9'/
      &       1X,29('-'),10('I----'),'-')
        XTRIAL=NTRIAL
        BIGREL=0.
C COMPUTE RELATIVE FREQUENCY FOR EACH CLASS INTERVAL, AND PLACE THE
C LARGEST RELATIVE FREQUENCY IN 'BIGREL'.
        DO 10 I=1,NFREQ
        RFREQS(I)=FREQS(I)/XTRIAL
    10 IF (RFREQS(I).GT.BIGREL) BIGREL=RFREQS(I)
C INITIALIZE CLASS INTERVAL CENTER AND CUMULATIVE RELATIVE FREQUENCY.
       CINTVL=CICTR
       CUMFRQ=BELOW/XTRIAL
        DO 60 NF=1,NFREQ
C CLEAR THE 50 CHARACTERS FOR THE BAR CORRESPONDING TO THIS CLASS INTERV
        DO 20 I=1,50
    20 BRCHRT(I)=BLANK
C 'XPLUS' AND 'NPLUS' ARE THE NUMBER OF PLUSES THAT WILL BE PLACED ON
C THE BAR FOR THIS CLASS INTERVAL.
        IF (NORMLZ) XPLUS=RFREQS(NF)/BIGREL*50.
        IF (.NOT.NORMLZ) XPLUS=RFREQS(NF)/0.02
        NPLUS=XPLUS
        IF (NPLUS.EQ.0) GO TO 40
C PLACE THE PLUSES ON THE BAR.
        DO 30 I=1,NPLUS
    30 BRCHRT(I)=PLUS
C CHECK FOR A FRACTION OF A PLUS.  IF IT EXISTS, ADD AN ASTERICK
C TO THE END OF THE BAR.
    40 IF (NPLUS.NE.XPLUS) BRCHRT(NPLUS+1)=ASTSK
C UPDATE THE CUMULATIVE RELATIVE FREQUENCY.
       CUMFRQ=CUMFRQ+RFREQS(NF)
C PRINT THE CLASS INTERVAL CENTER, CUMULATIVE RELATIVE FREQUENCY, RELATI
C FREQUENCY, AND BAR CHART FOR THIS CLASS INTERVAL.
        WRITE("    ",50) CINTVL,CUMFRQ,RFREQS(NF),BRCHRT
    50 FORMAT(F9.2,2F9.3,3X,'I',50A1)
C UPDATE TO THE NEXT CLASS INTERVAL CENTER.
    60 CINTVL=CINTVL+CIDEL
C PRINT THE BOTTOM BORDER AND LEAVE.
        IF (NORMLZ) WRITE("    ",70)
    70 FORMAT(1X,29('-'),'I',50('-'))
        IF (.NOT.NORMLZ) WRITE("    ",80)
    80 FORMAT(1X,29('-'),10('I----'),'-')
        RETURN
        END
C
```

```
      SUBROUTINE RETABL (NFREQ,NREPS)
      COMMON /RANGE/ NBASKL,NBASKH,NBASK
      COMMON /HIST/ FREQ(21,10),IBLANK(20),BLANK(50,10)
      INTEGER FREQ
      REAL RFREQ(21,10),CFREQ(6)
      WRITE("    ",101)
      WRITE("    ",102)
      WRITE("    ",103)
      WRITE("    ",104)
      WRITE("    ",105)
      XNREPS=NREPS
      DO 10 NBSK=NBASKL,NBASKH
      DO 10 NF=1,NFREQ
   10    RFREQ(NF,NBSK)=FREQ(NF,NBSK)/XNREPS
      DO 30 NBSK=NBASKL,NBASKH
         MIDDLE=NFREQ/2+1
         CTOT=RFREQ(MIDDLE,NBSK)
         DO 20 I=1,6
            CFREQ(I)=CTOT+RFREQ(MIDDLE-I,NBSK)+RFREQ(MIDDLE+I,NBSK)
   20       CTOT=CFREQ(I)
         WRITE("    ",106)
         WRITE("    ",107)NBSK,(CFREQ(I),I=1,6)
         WRITE("    ",106)
   30    WRITE("    ",105)
  101 FORMAT('-',25X,'RELATIVE FREQUENCY OF PERCENT OVERAWARD')
  102 FORMAT('-',11X,'NUMBER',10X,'PERCENT OVERAWARD INTERVAL')
  103 FORMAT(' ',13X,'OF')
  104 FORMAT(' ',11X,'BASKTS',5X,'+-1',7X,'+-2',7X,'+-3',7X,'+-4',7X,
     *          '+-5',7X,'+-6')
  105 FORMAT(' ',10X,69('-'))
  106 FORMAT(' ',10X,'|',6X,'||',6(9X,'|'))
  107 FORMAT(' ',10X,'|',2X,I2,2X,'||',6(2X,F5.3,2X,'|'))
      RETURN
      END
C
```

```
C
      SUBROUTINE PRICE(*)
      COMMON /PERSEL/ NBEP,RAND,IMBAL,NCODE
      LOGICAL RAND
      CALL PRINPU
      CALL PRINIT
      IF(RAND) GO TO 1
      CALL CHOICE($2)
    1 CALL RANDOM
    2 CALL BDORD
      CALL BASKT
      CALL SWAP
      CALL NEGTOT
      CALL LISTB($3)
      CALL STATS
      CALL OUTPUT
    3 CALL LISTP($4)
      CALL PRICOM
      CALL LSTPRC
      GO TO 3
    4 RETURN1
      END
C
C
      SUBROUTINE PRINPU
      CALL NUMBSK
      CALL NUMSUB
      CALL TYPSEL
      RETURN
      END
C
C
      SUBROUTINE NUMBSK
C   INPUTS NUMBER OF BASKETS INTO - NBASK.
C
      COMMON /RANGE/ NBASKL,NBASKH,NBASK
      COMMON /POP/ NBIDS,CHNUMP(12)
      COMMON /SUBPOP/ NSUPP,CHNUMS(12)
      INTEGER CHNUMP,CHNUMS
    1 WRITE("    ",2)
    2 FORMAT(' ENTER NUMBER OF BASKETS.')
      READ("    ",*) NBASK
      NBASKH=NBASK
      IF(2.LE.NBASK.AND.NBASK.LE.10) RETURN
      WRITE("    ",3) NBASK
    3 FORMAT(/,' *** ERROR *** NUMBER OF BASKETS SPECIFIED,'I4,
     &15X,'IS NOT BETWEEN 2 AND 10 INCLUSIVE.',/,' DO YOU WISH',
     &' TO REENTER NUMBER OF BASKETS? (YES OR NO)')
      CALL ANSWER($1)
      STOP
      END
C
```

```
12020C
12030         SUBROUTINE NEGTOT
12040C  CALCULATES TOTALS OF NEGOTIATED PRICES FOR EACH BASKET
12050C  AND PLACED INTO - BNEG(I).
12060C
12070         COMMON /RANGE/ NBASKL,NBASKH,NBASK
12080         COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
12090         COMMON /BACK/ BASKET(10,350),BASK1(10,160),BTOT(10),BNEG(10)
12100         COMMON /BIDORD/ BIDOR(160),NGROUP
12110         INTEGER BIDNUM,BIDTYP,BASKET,BASK1,BIDOR
12120         DO 1 I=1,NBACK
12130         BNEG(I)=0.
12140         DO 1 J=1,NGROUP
12150         IF(BASK1(I,J).EQ.-1) GO TO 1
12160         BNEG(I)=BNEG(I)+BIDNEG(BASK1(I,J))
12170       1 CONTINUE
12180         RETURN
12190         END
12200C
12210C
12220         SUBROUTINE LISTP(*)
12230         LOGICAL ARG
12240         WRITE("     ",1)
12241 PRINT,""
12250       1 FORMAT(/,' DO YOU WISH TO LIST BASKETS AND BASKET STATISTICS?'
12260        &,' (YES OR NO)')
12270         CALL ANSWR1(ARG)
12280         IF(ARG) RETURN
12290         RETURN1
12300         END
12310C
12320C
12330         SUBROUTINE STATS
12340         CALL PCTERB
12350         CALL MOMNTS
12360         RETURN
12370         END
12380C
12390C
12400         SUBROUTINE PCTERB
12410C   CALCULATES PER CENT ERROR BOUNDS.
12420C
12430         COMMON /RANGE/ NBASKL,NBASKH.NBACK
12440         COMMON /BACK/ BASKET(10,350),BASK1(10,160),BTOT(10),BNEG(10)
12450         COMMON /ERRBND/ MAXDIF,LOWBND,UPPBND,BCKEST(10),ESTDIF(10),
12460        &PCTEST(10)
12470         INTEGER BASKET,BASK1
12480         INTEGER BCK,ORDER1(10)
12490         REAL LOWBND,MAX,MIN,MAXDIF
12500         CALL ORDER(10,ORDER1,BTOT,NBACK)
12510         MAX=BTOT(ORDER1(NBASK))
12520         MIN=BTOT(ORDER1(1))
12530         MAXDIF=MAX-MIN
12540         TOTAL=0.
12550         TOTNEG=0.
```

```
      DO 1 BSK=1,NBASK
      TOTNEG=TOTNEG+BNEG(BSK)
    1 TOTAL=TOTAL+BTOT(BSK)
      LOWEND=100.*(TOTAL/NBASK/MAX-1.)
      UPREND=100.*(TOTAL/NBASK/MIN-1.)
      DO 2 BSK=1,NBASK
      DECREM=BNEG(BSK)/BTOT(BSK)
      DECBID=DECREM*(TOTAL-BTOT(BSK))
      BSKEST(BSK)=DECBID+BNEG(BSK)
      ESTDIF(BSK)=DECBID-TOTNEG+BNEG(BSK)
    2 PCTEST(BSK)=ESTDIF(BSK)/TOTNEG*100.
      RETURN
      END
C
C
      SUBROUTINE MOMNTS
C   CALCULATES MEAN, STD DEV FOR EACH BASKET AND POPULATION.
C
      COMMON /RANGE/ NBASKL,NBASKH,NBASK
      COMMON /SUBPOP/ NSUBP,CHNUMS(12)
      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
      COMMON /BASK/ BASKET(10,350),BASK1(10,160),BTOT(10),BNEG(10)
      COMMON /BIDORD/ BIDOR(160),NGROUP
      COMMON /MOM/ BSKMN(10),BSKVAR(10),POPMN,POPVAR,NBASK1(10),POPTOT
     &               ,POPNEG
      INTEGER CHNUMS,BIDNUM,BIDTYP,BASKET,BASK1,BIDOR
      INTEGER BSK
      DO 1 BSK=1,NBASK
    1 BSKVAR(BSK)=0.
      POPMN=0.
      POPVAR=0.
      POPNEG=0.
      DO 3 BSK=1,NBASK
      NBID=0
      POPNEG=POPNEG+BNEG(BSK)
      DO 2 M=1,NGROUP
      INDEX=BASK1(BSK,M)
      IF(INDEX.EQ.-1) GO TO 2
      NBID=NBID+1
      BSKVAR(BSK)=BSKVAR(BSK)+BIDPRI(INDEX)**2
    2 CONTINUE
      POPMN=POPMN+BTOT(BSK)
      POPVAR=POPVAR+BSKVAR(BSK)
      BSKMN(BSK)=BTOT(BSK)/NBID
      BSKVAR(BSK)=SQRT((BSKVAR(BSK)-BTOT(BSK)**2/NBID)/(NBID-1))
      NBASK1(BSK)=NBID
    3 CONTINUE
      POPVAR=SQRT((POPVAR-POPMN**2/NSUBP)/(NSUBP-1))
      POPTOT=POPMN
      POPMN=POPMN/NSUBP
      RETURN
      END
C
```

```
C
      SUBROUTINE OUTPUT
      CALL RSTATS
      CALL LSTBSK
      CALL DSTATS
      RETURN
      END
C
C
      SUBROUTINE RSTATS
C   PRINTS RUN STATISTICS.
C
      COMMON /POP/ NBIDS,CHNUMP(12)
      COMMON /SUBPOP/ NSUBP,CHNUMS(12)
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRBSK(250,12)
      COMMON /REPSEL/ NREP,RAND,IMBAL,NCODE
      INTEGER CHNUMP,CHNUMS,CHRNAM,CHRBSK
      LOGICAL RAND
      WRITE("     ",1) NBIDS
    1 FORMAT(///,' RUN STATISTICS',//,5X,'POPULATION SIZE = ',I3,/)
      IF(NCHAR.EQ.1) GO TO 3
      WRITE("     ",2) ((CHRNAM(I,J),J=1,2),CHNUMP(I),I=1,NCHAR)
    2 FORMAT(10X,2A4,' = ',I3)
    3 WRITE("     ",4) NSUBP
    4 FORMAT(' ',//,5X,'SUBPOPULATION SIZE = ',I3,/)
      GO TO (6,8,9), NCODE
    6 WRITE("     ",7)
    7 FORMAT(/,5X,'SELECTION METHOD:  SIMPLE RANDOM SAMPLING.',/)
      RETURN
    8 WRITE("     ",5)
    5 FORMAT(/,5X,'SELECTION METHOD:  STRATIFIED RANDOM SAMPLING-',/,
     &27X,'PROPORTION ALLOCATION',/)
      GO TO 12
    9 WRITE("     ",10)
   10 FORMAT(/,5X,'SELECTION METHOD:  STRATIFIED RANDOM SAMPLING-',/,
     &27X,'USER ALLOCATION',/)
   12 WRITE("     ",2) ((CHRNAM(I,J),J=1,2),CHNUMS(I),I=1,NCHAR)
      RETURN
      END
C
```

```
C
      SUBROUTINE LSTBSK
C  LISTS BASKETS.
C
      COMMON /RANGE/ NBASKL,NBASKH,NBASK
      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
      COMMON /BASK/ BASKET(10,350),BASK1(10,100),BTOT(10),BNEG(10)
      COMMON /BIDORD/ BIDOR(100),NGROUP
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRBSK(350,12)
      INTEGER BIDNUM,BIDTYP,BASKET,BASK1,BIDOR,CHRNAM,CHRBSK
      INTEGER BSK
      DO 10 BSK=1,NBASK
      WRITE("     ",1) BSK
  1   FORMAT(' ',//,5X,'BASKET NO.',2X,I2)
      IF(NCHAR.GT.1) GO TO 3
      WRITE("     ",2)
  2   FORMAT(' ',/,13X,'PROPOSAL NO.',10X,'BID PRICE',4X,'NEGOTIATED ',
     1'PRICE',/)
      GO TO 5
  3   WRITE("     ",4)
  4   FORMAT(' ',12X,'PROPOSAL NO.',10X,'BID PRICE',3X,'NEGOTIATED ',
     1'PRICE',2X,'CHARACTERISTIC',/)
  5   MINUS=0
      DO 8 N=1,NGROUP
      NN=N-MINUS
      INDEX=BASK1(BSK,N)
      IF(INDEX.EQ.-1) GO TO 9
      IF(NCHAR.EQ.1) GO TO 7
      WRITE("     ",6)NN,INDEX,BIDPRI(INDEX),BIDNEG(INDEX),
     1(CHRNAM(BIDTYP(INDEX),I),I=1,2)
  6   FORMAT(T10,I3,'.',T17,I4,T35,F10.2,T50,F10.2,T70,2A4)
      GO TO 8
  9   MINUS=MINUS+1
      GO TO 8
  7   WRITE("     ",6) N,INDEX,BIDPRI(INDEX),BIDNEG(INDEX)
  8   CONTINUE
 10   CONTINUE
      WRITE("     ",11)
 11   FORMAT(' ',5X,40('-'))
      RETURN
      END
C
```

```
C
C     SUBROUTINE BSTATS
C   PRINTS BASKET STATISTICS.
C
      COMMON /CURPOP/ NCURP,CHNUMS(12)
      COMMON /RANGE/ NBASKL,NBASKH,NBASK
      COMMON /BASK/ BASKET(10,350),BASK1(10,10 ),BTOT(10),BNEG(10)
      COMMON /ERRBND/ MAXDIF,LOWBND,UPRBND,BSKEST(10),ESTDIF(10),
     %PCTEST(10)
      COMMON /NCM/ BSKMN(10),BSKVAR(10),POPMN,POPVAR,NBASK1(10),POPTOT
     %            ,POPNEG
      INTEGER CHNUMS,BASKET,BASK1
      REAL LOWBND,MAX,MIN,MAXDIF
      WRITE("     ",1)
 1    FORMAT(' ',/,2X,'BASKET STATISTICS',/)
      WRITE("     ",2) MAXDIF
 2    FORMAT(T3,'DIFFERENCE BETWEEN MAX AND MIN BASKETS =',2X,F10.3)
      WRITE("     ",3) LOWBND,UPRBND
 3    FORMAT(T3,'MEAN ERROR BOUNDED BY ',F5.2,'% AND ',F5.2,'%.')
      WRITE("     ",4)
 4    FORMAT(T3,'BID PRICE MOMENTS:')
      WRITE("     ",5)
 5    FORMAT(/,3X,'NO.',5X,'TOTAL',7X,'TOTAL',20X,'POPUL',3X,'PERCENT',
     %/,'BASKET',2X,'BIDS',4X,'OF BIDS',4X,'NEGOT.',7X,'MEAN',
     %5X,'STD DEV',5X,'AWARD',4X,'ERROR',/)
      DO 6 N=1,NBASK
 6    WRITE("     ",7) N,NBASK1(N),BTOT(N),BNEG(N),BSKMN(N),BSKVAR(N),
     %BSKEST(N),PCTEST(N)
 7    FORMAT(2X,I2,4X,I3,3X,F10.3,2X,F10.3,2X,F9.2,2X,F9.2,2X,F10.3,1X,
     %F6.2)
      WRITE("     ",8) NCURP,POPTOT,POPNEG,POPMN,POPVAR
 8    FORMAT(/,1X,'POPUL',2X,I3,3X,F10.3,2X,F10.3,2X,F9.2,2X,
     %F9.2,//,2X,80('-'),/)
      RETURN
      END
      SUBROUTINE LISTP(*)
      COMMON /RANGE/ NBASKL,NBASKH,NBASK
      COMMON /PRICES/ BSKSEL,ESTIM(350),ERROR(350),PCTERR(350)
     %                ,ISPCHR(350),TOTBID,TOTNEG,TOTEST,TOTERR,MEAN
      INTEGER BSKSEL
      LOGICAL ARG
      WRITE("     ",1)
PRINT,""
    1 FORMAT(/,' DO YOU WISH TO MAKE A BASKET SELECTION FOR PRICING?',
     %' (YES OR NO)')
      CALL ANSWR1(ARG)
      IF(.NOT.ARG) RETURN1
    2 WRITE("     ",3) NBASK
PRINT,""
    3 FORMAT(' ENTER BASKET NUMBER BETWEEN 1 AND ',I2,' INCLUSIVE.')
      READ("     ",*) BSKSEL
      IF(1.LE.BSKSEL.AND.BSKSEL.LE.NBACK) RETURN
      WRITE("     ",4) BSKSEL
    4 FORMAT(/,' *** ERROR *** REQUESTED BASKET, ',I4,' IS NOT VALID.'
     %,/,15X,'DO YOU WISH TO REENTER? (YES OR NO)')
      CALL ANSWER(*2)
      STOP
      END
C
```

```
C
      SUBROUTINE PRICON
C  CALCULATES PRICES WHEN DECREMENT IS APPLIED TO OTHER BASKETS.
      COMMON /POP/ NBIDS,CHNUMP(12)
      COMMON /RANGE/ NBACKL,NBACKH,NBACK
      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
      COMMON /BACK/ BASKET(10,350),BACK1(10,100),ETOT(10),ENEG(10)
      COMMON /BIDORD/ BIDOR(100),NGROUP
      COMMON /PRICES/ BSKSEL,ESTIM(350),ERROR(350),PCTERR(350)
     ,                ,ISPCHR(350),TOTBID,TOTNEG,TOTEST,TOTERR,MEAN
      COMMON /TAB1/ TAB(350)
      LOGICAL TAB
      INTEGER BIDNUM,BIDTYP,BASKET,BACK1,BIDOR,BSKSEL
      REAL MEAN
      INTEGER ASTSK,BLANK
      DATA ASTSK/'*'/,BLANK/' '/
      DECREM=ENEG(BSKSEL)/ETOT(BSKSEL)
      TOTBID=0.
      TOTNEG=0.
      TOTEST=0.
      TOTERR=0.
      DO 10 I=1,NBIDS
   10 TAB(I)=.FALSE.
      DO 4 I=1,NBACK
      TOTBID=TOTBID+ETOT(I)
      TOTNEG=TOTNEG+ENEG(I)
      IF(I.EQ.BSKSEL) GO TO 2
      DO 1 J=1,NGROUP
      IF(BACK1(I,J).EQ.-1) GO TO 1
      INDEX=BACK1(I,J)
      TAB(INDEX)=.TRUE.
      ESTIM(INDEX)=DECREM*BIDPRI(INDEX)
      ISPCHR(INDEX)=BLANK
      ERROR(INDEX)=ESTIM(INDEX)-BIDNEG(INDEX)
      PCTERR(INDEX)=ERROR(INDEX)/BIDNEG(INDEX)*100.
      TOTERR=TOTERR+ERROR(INDEX)
      TOTEST=TOTEST+ESTIM(INDEX)
    1 CONTINUE
      GO TO 4
    2 DO 3 J=1,NGROUP
      IF(BACK1(I,J).EQ.-1) GO TO 3
      INDEX=BACK1(I,J)
      TAB(INDEX)=.TRUE.
      ESTIM(INDEX)=BIDNEG(INDEX)
      ISPCHR(INDEX)=ASTSK
      ERROR(INDEX)=ESTIM(INDEX)-BIDNEG(INDEX)
      PCTERR(INDEX)=ERROR(INDEX)/BIDNEG(INDEX)*100.
      TOTERR=TOTERR+ERROR(INDEX)
      TOTEST=TOTEST+ESTIM(INDEX)
    3 CONTINUE
    4 CONTINUE
      MEAN=TOTERR/TOTNEG*100.
      RETURN
      END
C
```

```
      SUBROUTINE LSTPRC
C LISTS PRICES.
      COMMON /POP/ NBIDS,CHNUMB(12)
      COMMON /SUBPOP/ NSUBP,CHNUMS(12)
      COMMON /BID/ BIDNUM(350),BIDPRI(350),BIDNEG(350),BIDTYP(350)
      COMMON /PRICES/ BSKSEL,ESTIM(350),ERROR(350),PCTERR(350)
     $               ,ISPCHR(350),TOTBID,TOTNEG,TOTEST,TOTERR,MEAN
      COMMON /TAB1/ TAB(350)
      LOGICAL TAB
      INTEGER CHNUMS,BSKSEL,BIDNUM,BIDTYP
      REAL MEAN
      WRITE("    ",1) BSKSEL
    1 FORMAT(//,' BASKET ',I2,' SELECTED.')
      WRITE("    ",2)
    2 FORMAT(/,' PROPOSAL',10X,'BID',3X,'NEGOTIATED',5X,'ESTIMATED'
     %,7X,'ERROR',5X,'PERCENT'/'  NUMBER',3(10X,'PRICE'),10X,'ERROR'
     %/' ')
      DO 4 I=1,NBIDS
      IF(.NOT.TAB(I)) GO TO 4
      WRITE("    ",3) I,BIDPRI(I),BIDNEG(I)
     %,ESTIM(I),ERROR(I),ISPCHR(I),PCTERR(I)
    3 FORMAT(I9,3F15.3,F12.3,A1,F11.2)
    4 CONTINUE
      WRITE("    ",5) TOTBID,TOTNEG,TOTEST,TOTERR,MEAN
    5 FORMAT(10X,3(6X,9('-')),5X,7('-')/'   TOTALS',3F15.3,F12.3/
     %' PERCENT ERROR IS',F6.2,'%.')
      RETURN
      END
```

```
BASSEL      13:34PDT    07/13/81

1 $OPTION PRTP
1000C  BASKET METHOD
1100C
1200C  ALGORITHM DESIGN AND IMPLEMENTATION - DR. STEPHEN L. BENZ
1300C                                         CLEMSON UNIVERSITY
1400C                                         CLEMSON, S.C. 29631
1500C
1600C  VERSION III, JUNE 1981.
1700C
1800C  THIS ALGORITHM SOLVES THE FOLLOWING PROBLEM.
1900C
2000C     GIVEN: (1)  A SET OF BIDPRICE DATA WITH EACH ITEM TYPED BY
2100C                 SOME OUTSIDE CRITERION (1 TO 12 TYPES).
2200C            (2)  NUMBER OF DESIRED BASKETS (2 TO 10 BASKETS).
2300C
2400C     PLACE THE BIDS INTO THE BASKETS TO OBTAIN THE FOLLOWING:
2500C
2600C            (1)  THE BIDS IN EACH BASKET SHOULD BE REPRESENTATIVE IN
2700C                 SIZE OF THE PARENT POPULATION.
2800C            (2)  THE COLLECTIVE BIDS OF ANY TYPE WITHIN EACH BASKET
2900C                 SHOULD BE REPRESENTATIVE OF THE PARENT TYPE POPULATION
3000C            (3)  THE TOTALS OF THE BASKETS SHOULD BE EQUAL.
3100C            (4)  THE NUMBER OF BIDS IN ANY BASKET SHOULD BE WITHIN
3200C                 ONE OF THE NUMBER IN ANY OTHER BASKET.
3300C            (5)  THE NUMBER OF BIDS OF ANY TYPE IN ANY BASKET SHOULD BE
3400C                 WITHIN ONE OF THE NUMBER OF BIDS OF THE SAME TYPE IN
3500C                 ANY OTHER BASKET.
3600C
3700C  THE ALGORITHM IS AS FOLLOWS.
3800C
3900C    NCHAR - THE NUMBER OF CHARACTER TYPES.
4000C    NBASK - THE NUMBER OF BASKETS DESIRED.
4100C
4200C    (1)  THE BIDS ARE PLACED INTO NCHAR BASKETS BY CHARACTER TYPES AND
4300C         ORDERED WITHIN EACH BASKET BY SIZE FROM LARGEST TO SMALLEST.
4400C    (2)  THE BIDS ARE PLACED FROM THESE NCHAR BASKETS INTO THE GIVEN
4500C         NBASK BASKETS BY GROUPS OF NBASK, ALL OF A SINGLE TYPE.  THE
4600C         TYPE TO BE PLACED IS DETERMINED TO BE THE ONE HAVING THE
4700C         LARGEST AVERAGE OF THE NEXT NBASK BIDS.  THESE BIDS ARE THEN
4800C         PLACED INTO THE BASKETS, THE SMALLEST BID GOING TO THE BASKET
4900C         WITH THE LARGEST TOTAL, THE NEXT SMALLEST BID GOING TO THE
5000C         BASKET WITH THE NEXT LARGEST TOTAL, AND SO ON UNTIL ALL THE
5100C         BIDS OF THAT GROUPING HAVE BEEN PLACED.  THIS PROCESS IS
5200C         REPEATED UNTIL ALL THE BIDS HAVE BEEN PLACED.
5300C    (3)  SINCE THE ABOVE PLACEMENT OF THE BIDS DOES NOT GUARANTEE
5400C         EQUAL BASKET TOTALS, BIDS OF SIMILAR TYPE ARE SWAPPED
5500C         BETWEEN THE LARGEST AND THE SMALLEST BASKETS UNTIL SWAPPING
5600C          WILL NO LONGER IMPROVE THE BALANCE OF THE BASKET TOTALS.
5700C
```

```
580C
590C   MAIN PROGRAM
600        CALL DSPLAY
610        CALL INPUT
620    1   CALL CHBASK
630        CALL BSKORD
640        CALL BIDORD
650        CALL BASKT
660        CALL SWAP
670    2   CALL STATS
680        CALL OUTPUT
690        CALL CONT($1,$2)
700        STOP
710        END
720C
730C
740        SUBROUTINE DSPLAY
750        WRITE("     ",1)
760      1 FORMAT(' ',76('*')/' *',74X,'*'
770       &/' *',30X,'BASKET METHOD',31X,'*'/
780       &' *',24X,'VERSION III - JUNE 1981',27X,'*'/' *',74X
790       &,'*'/' *',14X,'THEORETICAL DEVELOPMENT - DR. K. T. WALLENIUS',
800       &15X,'*'/' *',10X,'ALGORITHM DESIGN AND IMPLEMENTATION -',
810       &' DR. STEPHEN BENZ',10X,'*'/' *',74X,'*'/' *',28X,'CLEMSON',
820       &' UNIVERSITY',28X,'*'/' *',31X,'CLEMSON S.C.',31X,'*'/' *',
830       &74X,'*'/' ',76('*'))
840        RETURN
850        END
860C
870        SUBROUTINE INPUT
880        INTEGER CHRNAM,CHRNUM,CHRBSK
890        COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRBSK(12,300)
900        CALL NUMBSK
910        CALL NUMCHR
920        IF(NCHAR.EQ.1) GO TO 1
930        CALL CHRTYP
940    1   CALL BIDS
950        RETURN
960        END
970C
```

```
980C
990       SUBROUTINE NUMBSK
1000C   READS NUMBER OF BASKETS INTO - NBASK.
1010C
1020       INTEGER ANSWR,YES,NO
1030       INTEGER BASKET,BASK1
1040       COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTCT(10),NGROUP
1050       DATA  YES/'Y'/,NO/'N'/
1060  1    WRITE("     ",2)
1061 PRINT,""
1070  2    FORMAT(' ENTER NUMBER OF BASKETS.')
1080       READ("    ",*) NBASK
1090       IF(NBASK.GE.2.AND.NBASK.LE.10) RETURN
1100       WRITE("     ",3) NBASK
1110  3    FORMAT(' ',/,' *** ERROR ***  NUMBER OF BASKETS SPECIFIED,',I4,','
1120       ,/,15X,'   IS NOT BETWEEN 2 AND 10 INCLUSIVE.')
1130       WRITE("     ",4)
1131 PRINT,""
1140  4    FORMAT(' DO YOU WISH TO REENTER NUMBER OF BASKETS? (YES OR NO)')
1150  5    READ("    ",6) ANSWR
1160  6    FORMAT(A1)
1170       IF(ANSWR.EQ.YES) GO TO 1
1180       IF(ANSWR.EQ.NO) STOP
1190       WRITE("     ",7)
1191 PRINT,""
1200  7    FORMAT('REENTER "YES OR NO"')
1210       GO TO 5
1220       END
1230C
1240C
1250       SUBROUTINE NUMCHR
1260C   READS NUMBER OF CHARACTER TYPES INTO - NCHAR.
1270C
1280       INTEGER ANSWR,YES,NO
1290       INTEGER CHRNAM,CHRNUM,CHRBSK
1300       COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRBSK(12,300)
1310       DATA  YES/'Y'/,NO/'N'/
1320  1    WRITE("     ",2)
1321 PRINT,""
1330  2    FORMAT(' ',/,' ENTER NUMBER OF CHARACTER TYPES.')
1340       READ("    ",*) NCHAR
```

```
1350          IF(NCHAR.GT.0.AND.NCHAR.LE.12) RETURN
1360          WRITE("     ",3) NCHAR
1370    3     FORMAT(' ',/,' *** ERROR ***  NUMBER OF CHARACTER TYPES,',I3,',',
1380         &7X,' IS NOT BETWEEN 1 AND 12 INCLUSIVE.')
1390          WRITE("     ",4)
1391 PRINT,""
1400    4     FORMAT(' DO YOU WISH TO REENTER NUMBER OF CHARACTER TYPES?',
1410         &' (YES OR NO)')
1420    5     READ("     ",6) ANSWR
1430    6     FORMAT(A1)
1440          IF(ANSWR.EQ.YES) GO TO 1
1450          IF(ANSWR.EQ.NO) STOP
1460          WRITE("     ",7)
1461 PRINT,""
1470    7     FORMAT('REENTER "YES OR NO"')
1480          GO TO 5
1490          END
1500C
1510C
1520          SUBROUTINE CHRTYP
1530C   READS CHARACTER TYPES INTO - CHRNAM(N,K), K=1,2.
1540C
1550          INTEGER ANSWR,YES,BLANK,NO
1560          INTEGER CHRNAM,CHRNUM,CHRBSK
1570          COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRBSK(12,300)
1580          DATA  YES/'Y'/,BLANK/'    '/,NO/'N'/
1590    1     WRITE("     ",2) NCHAR
1591 PRINT,""
1600    2     FORMAT(' ',/,' ENTER',I3,', 8 - CHARACTER ID''S FOR CHARACTER',
1610         &' TYPES, ONE PER LINE.')
1620          DO 3 J=1,NCHAR
1630    3     READ ("     ",4) (CHRNAM(J,K),K=1,2)
1640    4     FORMAT(2A4)
1650          DO 5 N=1,NCHAR
1660          IF(CHRNAM(N,1).EQ.BLANK.AND.CHRNAM(N,2).EQ.BLANK) GO TO 6
1670    5     CONTINUE
1680          RETURN
1690    6     WRITE("     ",7) NCHAR
1700    7     FORMAT(' ',/,' *** ERROR ***  NUMBER OF CHARACTER TYPES DOES',/,
1710         &17X,' NOT EQUAL',I3,', OR DATA NOT ENTERED IN PROPER FORMAT.')
1720          WRITE("     ",8)
1721 PRINT,""
1730    8     FORMAT(' DO YOU WISH TO REENTER CHARACTER TYPES? (YES OR NO)')
1740    9     READ("     ",10) ANSWR
1750    10    FORMAT(A1)
1760          IF(ANSWR.EQ.YES) GO TO 1
1770          IF(ANSWR.EQ.NO) STOP
1780          WRITE("     ",11)
1781 PRINT,""
1790    11    FORMAT('REENTER "YES OR NO"')
1800          GO TO 9
1810          END
1820C
```

```
C
      SUBROUTINE BIDS
C  READS DATA.
C       BIDNUM(N) - BID PROPOSAL NUMBER.
C       BIDPRI(N) - BID PRICE.
C       BIDTYP(N) - INTEGER VALUES FROM 1 TO 12 CORRESPONDING
C                   TO TYPE OF BID.
      INTEGER BDTYP1,BDTYP2,ANSWR,YES,NO
STRING FILENAM
      INTEGER BIDNUM,BIDTYP,BIDOR
      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
      INTEGER CHRNAM,CHRNUM,CHRESK,PROP
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRESK(12,300)
      DATA  YES/'Y'/,NO/'N'/
101 PRINT,"ENTER FILENAME"
INPUT, FILENAM
OPEN(FILENAM,STATUS="OLD",UNIT=9,ERR=110)
GOTO 100
110 PRINT,"FILE NOT FOUND"
GOTO 101
100   NERR=0
      IF(NCHAR.GT.1) GO TO 2
      DO 1 NBIDS=1,300
      BIDNUM(NBIDS)=NBIDS
  1   READ(9,*,END=11) BIDPRI(NBIDS)
      GO TO 10
  2   DO 9 NBIDS=1,300
      BIDNUM(NBIDS)=NBIDS
  3   READ(9,*,END=11) BIDPRI(NBIDS),BIDTYP(NBIDS)
      IF(1.LE.BIDTYP(NBIDS).AND.BIDTYP(NBIDS).LE.NCHAR) GO TO 9
      PROP=BIDNUM(NBIDS)+NERR
      WRITE("     ",4) PROP,BIDTYP(NBIDS)
  4   FORMAT(' ',/,' *** ERROR ENCOUNTERED ON PROPOSAL',I4,'.'/
     &'     THE TYPE OF PROPOSAL ',I3,' DOES NOT MATCH WITH ANY OF',
     &' THOSE SUBMITTED.')
      NERR=NERR+1
      WRITE("     ",5)
PRINT,""
  5   FORMAT(' DO YOU WISH TO OMIT THIS PROPOSAL FROM THE ANALYSIS',
     &' AND CONTINUE? (YES OR NO)')
  6   READ("     ",7) ANSWR
  7   FORMAT(A1)
      IF(ANSWR.EQ.YES) GO TO 3
      IF(ANSWR.EQ.NO) STOP
      WRITE("     ",8)
PRINT,""
  8   FORMAT('REENTER "YES OR NO"')
      GO TO 6
  9   CONTINUE
  10  NBIDS=301
  11  NBIDS=NBIDS-1
      RETURN
      END
C
C
```

```
C
C
        SUBROUTINE CHBASK
C   PUTS BIDS INTO BASKETS - CHRBSK(N,K) - BY TYPE.
C          CHRNUM(N) - NUMBER OF BIDS OF CHARACTER TYPE N.
C
        INTEGER CHARA,ANSWR,YES,NO
        INTEGER BIDNUM,BIDTYP,BIDOR
        COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
        INTEGER BASKET,BASK1
        COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
        INTEGER CHRNAM,CHRNUM,CHRBSK
        COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRBSK(12,300)
        DATA YES/'Y'/,NO/'N'/
        IF(NCHAR.EQ.1) GO TO 9
        DO 1 CHARA=1,NCHAR
   1    CHRNUM(CHARA)=0
        DO 2 NBID=1,NBIDS
        INDEX=BIDTYP(NBID)
        CHRNUM(INDEX)=CHRNUM(INDEX)+1
   2    CHRBSK(INDEX,CHRNUM(INDEX))=NBID
        DO 8 CHARA=1,NCHAR
        IF(CHRNUM(CHARA).GE.NBASK) GO TO 8
        WRITE("     ",3) CHRNAM(CHARA,1),CHRNAM(CHARA,2),CHRNUM(CHARA),NBASK
   3    FORMAT(' ** WARNING ** THE NUMBER OF BIDS OF THE TYPE, ',2A4,','
       &,/,'                  TOTALS ONLY, ',I3,', WHICH IS LESS THAN'
       &,/,'                  THE NUMBER OF BASKETS,',I3,', TO BE FORMED.')
        WRITE("     ",4)
 PRINT,""
   4    FORMAT(' DO YOU WISH TO CONTINUE? (YES OR NO)')
   5    READ("     ",6) ANSWR
   6    FORMAT(A1)
        IF(ANSWR.EQ.YES) GO TO 8
        IF(ANSWR.EQ.NO) STOP
        WRITE("     ",7)
 PRINT,""
   7    FORMAT('REENTER "YES OR NO"')
        GO TO 5
   8    CONTINUE
        RETURN
   9    CHRNUM(1)=NBIDS
        DO 10 NBID=1,NBIDS
  10    CHRBSK(1,NBID)=NBID
        RETURN
        END
C
```

```
2730C
2740        SUBROUTINE BSKORD
2750C   PUTS BIDS WITHIN EACH CHARACTER TYPE BASKET IN ORDER
2760C   FROM LARGEST TO SMALLEST.
2770C
2780        INTEGER CHARA,FIRST,RECENT,TEMP
2790        INTEGER BIDNUM,BIDTYP,BIDOR
2800        COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
2810        INTEGER CHRNAM,CHRNUM,CHRBSK
2820        COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRBSK(12,300)
2830        COMMON /PLACE1/IND1
2840        IND1=NCHAR
2850        DO 5 CHARA=1,NCHAR
2860        RECENT=1
2870        LAST=CHRNUM(CHARA)
2880   1    FIRST=RECENT+1
2890        DO 2 J=FIRST,LAST
2900        JM1=J-1
2910        IF(BIDPRI(CHRBSK(CHARA,JM1)).GE.BIDPRI(CHRBSK(CHARA,J))) GO TO 2
2920        RECENT=JM1
2930        TEMP=CHRBSK(CHARA,JM1)
2940        CHRBSK(CHARA,JM1)=CHRBSK(CHARA,J)
2950        CHRBSK(CHARA,J)=TEMP
2960   2    CONTINUE
2970        IF(RECENT+1.EQ.FIRST) GO TO 5
2980        LAST=RECENT
2990        J=LAST
3000   3    JM1=J-1
3010        IF(BIDPRI(CHRBSK(CHARA,JM1)).GE.BIDPRI(CHRBSK(CHARA,J))) GO TO 4
3020        RECENT=J
3030        TEMP=CHRBSK(CHARA,JM1)
3040        CHRBSK(CHARA,JM1)=CHRBSK(CHARA,J)
3050        CHRBSK(CHARA,J)=TEMP
3060   4    J=J-1
3070        IF(J.GE.FIRST) GO TO 3
3080        IF(RECENT.LT.LAST) GO TO 1
3090   5    CONTINUE
3100        RETURN
3110        END
3120C
3130C
3140        SUBROUTINE BIDORD
3150C   DETERMINES ORDER IN WHICH BIDS WILL BE PLACED IN GROUPS OF NBASK.
3160C   CREATES - BIDOR(N) - WHICH IS THE INDEX OF THE CHARACTERISTIC
3170C   OF THE NTH BID GROUPING.
3180C        NGROUP - NUMBER OF SUCH GROUPINGS.
3190C
3200        INTEGER CHARA
3210        INTEGER BIDNUM,BIDTYP,BIDOR
3220        COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
3230        INTEGER BASKET,BASK1
3240        COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
3250        INTEGER CHRNAM,CHRNUM,CHRBSK
3260        COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRBSK(12,300)
3270        COMMON /AVGORD/AVG(12),NCHR(12),NUMBID,NCHAR1,MAX
```

```
      NCHAR1=NCHAR
      NGROUP=0
      NUMBID=0
      IF(NCHAR.EQ.1) GO TO 4
      DO 1 CHARA=1,NCHAR
    1 NCHR(CHARA)=0
      DO 3 N=1,400
      IF(NUMBID.GE.NBIDS) RETURN
      CALL MAXAVG
      NGROUP=NGROUP+1
      BIDOR(NGROUP)=MAX
    3 CONTINUE
      RETURN
    4 NGROUP=NBIDS/NBASK
      IF(NBASK*NGROUP.NE.NBIDS) NGROUP=NGROUP+1
      DO 5 NGROU=1,NGROUP
    5 BIDOR(NGROU)=1
      NCHR(1)=NBIDS
      RETURN
      END
C
C
      SUBROUTINE MAXAVG
C   CALCULATES AVERAGE OF THE NEXT NBASK BIDS OF EACH CHARACTERISTIC
C   TYPE, AND RETURNS THE INDEX OF THE CHARACTERISTIC TYPE HAVING THE
C   LARGEST AVERAGE.
C        NCHR(N) - INDEX OF POSITION TO BEGIN AVERAGE OF BIDS FOR
C                  CHARACTERISTIC TYPE N.
C
      INTEGER START,END
      INTEGER CHRNAM,CHRNUM,CHRBSK
      INTEGER BIDNUM,BIDTYP,BIDOR
      INTEGER BASKET,BASK1
      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRBSK(12,300)
      COMMON /AVGORD/AVG(12),NCHR(12),NUMBID,NCHAR1,MAX
      MAX=1
      DO 2 N=1,NCHAR1
      AVG(N)=0
      START=NCHR(N)+1
      END=NCHR(N)+NBASK
      IF(END.GT.CHRNUM(N)) END=CHRNUM(N)
      IF(START.GT.END) GO TO 2
      DO 1 J=START,END
    1 AVG(N)=AVG(N)+BIDPRI(CHRBSK(N,J))
      IF(START.EQ.END) GO TO 3
      AVG(N)=AVG(N)/(END-START)
    3 IF(AVG(N).LT.AVG(MAX)) GO TO 2
      MAX=N
      MAXBDS=END-START+1
    2 CONTINUE
      NUMBID=NUMBID+MAXBDS
      NCHR(MAX)=NCHR(MAX)+MAXBDS
      RETURN
      END
C
```

```
C
C
      SUBROUTINE BASKT
C  PLACES BIDS INTO BASKETS ACCORDING TO BIDOR AND BASKET TOTALS.
C        BTOT(N) - TOTAL OF BIDS IN BASKET N.
C
      REAL HIDE(10)
      INTEGER START(12),UP1(10),ORJ,ORDER1(10),ORDER2(10),BIDORI
      INTEGER CHRNAM,CHRNUM,CHRBSK
      INTEGER BIDNUM,BIDTYP,BIDOR
      INTEGER BASKET,BASK1
      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRBSK(12,300)
      COMMON /AVGORD/AVG(12),NCHR(12),NUMBID,NCHAR1,MAX
      DO 1 N=1,NBASK
      HIDE(N)=0
      BTOT(N)=0
 1    UP1(N)=0
      DO 20 N=1,NCHAR
 20   START(N)=0
      DO 6 I=1,NGROUP
      BIDORI=BIDOR(I)
      NUMLEF=NCHR(BIDORI)-START(BIDORI)
      CALL ORDER(ORDER1,BTOT,NBASK)
      IF(NUMLEF.LT.NBASK) GO TO 3
      DO 2 J=1,NBASK
      ORJ=ORDER1(J)
      INDEX=CHRBSK(BIDORI,START(BIDORI)+J)
      PRICE=BIDPRI(INDEX)
      BASKET(ORJ,I)=INDEX
 2    BTOT(ORJ)=BTOT(ORJ)+PRICE
      START(BIDORI)=START(BIDORI)+NBASK
      GO TO 6
 3    CALL UPONE(ORDER1,ORDER2,NBASK,UP1,NUMLEF)
      AVGE=0.
      DO 4 J=1,NUMLEF
      ORJ=ORDER2(J)
      INDEX=CHRBSK(BIDORI,START(BIDORI)+J)
      PRICE=BIDPRI(INDEX)
      BASKET(ORJ,I)=INDEX
      AVGE=AVGE+PRICE
 4    BTOT(ORJ)=BTOT(ORJ)+PRICE
      AVGE=AVGE/NUMLEF
      NMLEF1=NUMLEF+1
      DO 5 J=NMLEF1,NBASK
      ORJ=ORDER2(J)
C
```

```
C   WHEN NUMBER OF BIDS LEFT OF ANY PARTICULAR CHARACTERISTIC TYPE
C   IS LESS THAN NUMBER OF BASKETS, THE AVERAGE OF THESE BIDS IS
C   CALCULATED, THE BIDS ARE PLACED INTO THE BASKETS, AND THE
C   REMAINING UNFILLED BASKETS RECEIVE THE AVERAGE OF THE BIDS.
C   (SINCE BASKET(N,K) CONTAINS ONLY INDICES OF PROPOSALS AND NOT
C    ACTUAL BID PRICES, THE UNFILLED BASKETS RECEIVE THE VALUE -1
C    INTO BASKET(N,K).)
C        HIDE(N) - ACCUMULATES THE PHANTOM AVERAGES IN BASKET N
C                  FOR LATER CORRECTION OF BTOT(N).
C
        BASKET(ORJ,I)=-1
        HIDE(ORJ)=HIDE(ORJ)+AVGE
   5    BTOT(ORJ)=BTOT(ORJ)+AVGE
   6    CONTINUE
        DO 7 J=1,NBASK
   7    BTOT(J)=BTOT(J)-HIDE(J)
        RETURN
        END
C
C
        SUBROUTINE ORDER(ORDER1,BTOT,NBASK)
C   ORDERS THE BASKET INDICES INTO - ORDER1 - FROM SMALLEST BASKET
C   TO LARGEST BASKET.
C
        INTEGER ORDER1(10)
        REAL BTOT(10)
        DO 1 N=1,NBASK
   1    ORDER1(N)=N
        NBASK1=NBASK-1
        DO 3 N=1,NBASK1
        MIN=N
        NBASK2=N+1
        DO 2 K=NBASK2,NBASK
        IF(BTOT(ORDER1(MIN)).LE.BTOT(ORDER1(K))) GO TO 2
        ITEMP=ORDER1(MIN)
        ORDER1(MIN)=ORDER1(K)
        ORDER1(K)=ITEMP
   2    CONTINUE
   3    CONTINUE
        RETURN
        END
C
C
        SUBROUTINE UPONE(ORDER1,ORDER2,NBASK,UP1,NUMLEF)
C   DETERMINES WHICH BASKETS HAVE ONE MORE BID THAN OTHERS AND RETURNS
C   - ORDER2 - THE NEW ORDER OF BASKET INDICES REFLECTING SMALLEST TO
C   LARGEST AND NUMBER OF BIDS IN BASKETS.
C
        INTEGER ORDER1(10),ORDER2(10),UP1(10),ENDJ,STARTJ
        STARTJ=0
        DO 1 I=1,NBASK
        IF(UP1(ORDER1(I)).EQ.1) GO TO 1
        STARTJ=STARTJ+1
        ORDER2(STARTJ)=ORDER1(I)
   1    CONTINUE
```

```
      IF(STARTJ.NE.0) GO TO 3
      DO 2 J=1,NBASK
2     ORDER2(J)=ORDER1(J)
      GO TO 8
3     IF(STARTJ.EQ.NBASK) GO TO 8
      ENDJ=STARTJ
      DO 4 I=1,NBASK
      IF(UP1(ORDER1(I)).EQ.0) GO TO 4
      ENDJ=ENDJ+1
      ORDER2(ENDJ)=ORDER1(I)
4     CONTINUE
5     IF(STARTJ.GE.NUMLEF) GO TO 10
      DO 6 J=1,NUMLEF
      IF(UP1(ORDER2(J)).EQ.0) UP1(ORDER2(J))=0
6     IF(UP1(ORDER2(J)).EQ.1) UP1(ORDER2(J))=1
      NMLEF1=NUMLEF+1
      DO 7 J=NMLEF1,NBASK
7     UP1(ORDER2(J))=0
      RETURN
8     NMLEF1=NUMLEF+1
      DO 9 J=NMLEF1,NBASK
9     UP1(ORDER2(J))=0
10    DO 11 J=1,NUMLEF
11    UP1(ORDER2(J))=1
      RETURN
      END
C
C
      SUBROUTINE SWAP
C  PERFORMS A BID SWAP BETWEEN BASKETS TO BALANCE BASKET TOTALS.
C     LGBSK - INDEX OF LARGEST BASKET WITH RESPECT TO BASKET TOTALS.
C     SMBSK - INDEX OF SMALL BASKDET.
C
      INTEGER ORDER1(10),BSK,SWTCH1,SWTCH2
      INTEGER BIDNUM,BIDTYP,BIDOR
      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
      INTEGER BASKET,BASK1
      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
      INTEGER CHRNAM,CHRNUM,CHRBSK
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRBSK(12,300)
      INTEGER LGBSK,LGIND,SMBSK,SMIND,STARTC,CHAR
      COMMON /SWAP1/LGBSK,LGIND,SMBSK,SMIND,STARTC(12),CHAR
      CALL SETUP
1     CALL ORDER(ORDER1,BTOT,NBASK)
      DO 3 N=2,NBASK
      LGBSK=ORDER1(NBASK-N+2)
      NBASK1=NBASK-1
      DO 2 K=1,NBASK1
      SMBSK=ORDER1(K)
      DIFF=BTOT(LGBSK)-BTOT(SMBSK)
      IF(DIFF.EQ.0.) GO TO 3
      CALL DIFFNT(SWTCH2,DIFF)
      IF(SMIND.NE.0) GO TO 4
    2 CONTINUE
    3 CONTINUE
      RETURN
    4 DIFF=BIDPRI(BASK1(LGBSK,LGIND))-BIDPRI(BASK1(SMBSK,SMIND))
      BTOT(LGBSK)=BTOT(LGBSK)-DIFF
      BTOT(SMBSK)=BTOT(SMBSK)+DIFF
      CALL PLACE
      GO TO 1
      END
C
C
      SUBROUTINE SETUP
C  INITIALIZES - BASK1(N,K) - TO BE THE SAME AS - BASKET(N,K) - BUT
C  WITH BIDS PLACED TOGETHER WITHIN BASKETS BY CHARACTERISTIC TYPE.
C     NOTE: BIDS WILL ALSO BE IN ORDER FROM LARGEST TO SMALLEST
C           WITHIN CHARACTER GROUPINGS SINCE BASKET WAS ORDERED.
C
      INTEGER COUNT(12),POS,BSK
      INTEGER BIDNUM,BIDTYP,BIDOR
      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
      INTEGER BASKET,BASK1
      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
      INTEGER CHRNAM,CHRNUM,CHRBSK
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRBSK(12,300)
      INTEGER LGBSK,LGIND,SMBSK,SMIND,STARTC,CHAR
      COMMON /SWAP1/LGBSK,LGIND,SMBSK,SMIND,STARTC(12),CHAR
      DO 1 N=1,NCHAR
    1 COUNT(N)=0
      DO 2 N=1,NGROUP
      NUMB=BIDOR(N)
    2 COUNT(NUMB)=COUNT(NUMB)+1
      STARTC(1)=1
      DO 3 N=2,NCHAR
      STARTC(N)=COUNT(N-1)+STARTC(N-1)
    3 COUNT(N-1)=0
      COUNT(NCHAR)=0
      DO 5 N=1,NGROUP
      NUMB=BIDOR(N)
      POS=STARTC(NUMB)+COUNT(NUMB)
      DO 4 BCK=1,NBACK
    4 BASK1(LCK,POS)=BASKET(BCK,N)
    5 COUNT(NUMB)=COUNT(NUMB)+1
      RETURN
      END
C
C
      SUBROUTINE DIFFNT(SWTCH2,DIFF)
C  DETERMINES IF/WHICH BID SWAPS FOR GIVEN BASKETS WILL PRODUCE A
C  BETTER BALANCE.
C        LGIND - INDEX OF BID TO BE SWAPPED OUT OF LARGE BASKET.
C        SMIND - INDEX OF BID TO BE SWAPPED OUT OF SMALL BASKET.
C
      INTEGER SWTCH1,SWTCH2,STRT,STP,CHARA
      INTEGER BIDNUM,BIDTYP,BIDOR
      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
      INTEGER BASKET,BASK1
      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
      INTEGER CHRNAM,CHRNUM,CHRBSK
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRBSK(12,300)
      INTEGER LGBSK,LGIND,SMBSK,SMIND,STARTC,CHAR
      COMMON /SWAP1/LGBSK,LGIND,SMBSK,SMIND,STARTC(12),CHAR
      SMIND=0
      DIFF1=0
      DO 3 CHARA=1,NCHAR
      STRT=STARTC(CHARA)
      STP=NGROUP
      IF(CHARA.LT.NCHAR)STP=STARTC(CHARA+1)-1
      DO 2 SWTCH1=STRT,STP
      IND1=BASK1(LGBSK,SWTCH1)
      IF(IND1.EQ.-1) GO TO 2
      PRICE1=BIDPRI(IND1)
      DO 1 SWTCH2=STRT,STP
      IND2=BASK1(SMBSK,SWTCH2)
      IF(IND2.EQ.-1.) GO TO 1
      PRICE2=BIDPRI(IND2)
      DIFF2=PRICE1-PRICE2
      IF(DIFF2.LE.0.) GO TO 1
      IF(DIFF2.GE.DIFF) GO TO 2
      IF(ABS(DIFF2-DIFF/2.).GE.ABS(DIFF1-DIFF/2.)) GO TO 2
      DIFF1=DIFF2
      LGIND=SWTCH1
      SMIND=SWTCH2
      CHAR=CHARA
      IF(DIFF2.EQ.DIFF/2.) RETURN
    1 CONTINUE
    2 CONTINUE
    3 CONTINUE
      RETURN
      END
C
```

```
C
      SUBROUTINE PLACE
C   SWAPS BIDS.
C
      INTEGER LGBSK,LGIND,SMBSK,SMIND,STARTC,CHAR
      COMMON /SWAP1/LGBSK,LGIND,SMBSK,SMIND,STARTC(12),CHAR
      INTEGER BIDNUM,BIDTYP,BIDCR
      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDCR(160)
      INTEGER BASKET,BASK1
      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
      TEMP=BASK1(LGBSK,LGIND)
      BASK1(LGBSK,LGIND)=BASK1(SMBSK,SMIND)
      BASK1(SMBSK,SMIND)=TEMP
      CALL PUT(LGBSK,LGIND,CHAR)
      CALL PUT(SMBSK,SMIND,CHAR)
      RETURN
      END
C
C
      SUBROUTINE PUT(BSK,IND,CHAR1)
C   MOVES 'NEW BID' INTO PROPER ORDERED POSITION WITHIN CHARACTER TYPE
C   GROUPING WITHIN BASKET.
C
      INTEGER BIDNUM,BIDTYP,BIDCR
      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDCR(160)
      INTEGER BASKET,BASK1
      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
      INTEGER CHRNAM,CHRNUM,CHRBSK
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRBSK(12,300)
      INTEGER LGBSK,LGIND,SMBSK,SMIND,STARTC,CHAR
      COMMON /SWAP1/LGBSK,LGIND,SMBSK,SMIND,STARTC(12),CHAR
      INTEGER BSK,CHAR1,STRT,STP
      IF(IND.EQ.STARTC(CHAR1)) GO TO 2
      IF(BIDPRI(BASK1(BSK,IND)).LE.BIDPRI(BASK1(BSK,IND-1))) GO TO 2
      STRT=STARTC(CHAR1)
      STP=IND-1
      DO 1 K=STRT,STP
      KK=STP+STRT-K+1
      KK1=KK-1
      IF(BIDPRI(BASK1(BSK,KK)).LE.BIDPRI(BASK1(BSK,KK1))) RETURN
      TEMP=BASK1(BSK,KK)
      BASK1(BSK,KK)=BASK1(BSK,KK1)
      BASK1(BSK,KK1)=TEMP
    1 CONTINUE
      RETURN
    2 STRT=IND
      IF(STRT.EQ.NGROUP) RETURN
      STP=NGROUP-1
      IF(CHAR1.LT.NCHAR) STP=STARTC(CHAR1+1)-2
      IF(STRT.GT.STP) RETURN
      DO 3 K=STRT,STP
      IF(BASK1(BSK,K+1).EQ.-1) RETURN
      IF(BIDPRI(BASK1(BSK,K)).GE.BIDPRI(BASK1(BSK,K+1))) RETURN
      TEMP=BASK1(BSK,K)
      BASK1(BSK,K)=BASK1(BSK,K+1)
      BASK1(BSK,K+1)=TEMP
    3 CONTINUE
      RETURN
      END
C
```

```
C
      SUBROUTINE STATS
      CALL PCTERR
      CALL MOMNTS
      RETURN
      END
C
C
      SUBROUTINE PCTERR
C   CALCULATES PER CENT ERROR BOUNDS.
C
      INTEGER BSK,ORDER1(10)
      REAL LOWEND,MAX,MIN,MAXDIF
      COMMON /ERRBND/ MAXDIF,LOWEND,UPRBND
      INTEGER BASKET,BASK1
      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
      CALL ORDER(ORDER1,BTOT,NBASK)
      MAX=BTOT(ORDER1(NBASK))
      MIN=BTOT(ORDER1(1))
      MAXDIF=MAX-MIN
      TOTAL=0.
      DO 1 BSK=1,NBASK
 1    TOTAL=TOTAL+BTOT(BSK)
      LOWEND=100.*(TOTAL/NBASK/MAX-1.)
      UPRBND=100.*(TOTAL/NBASK/MIN-1.)
      RETURN
      END
C
C
      SUBROUTINE MOMNTS
C   CALCULATES MEAN, STD DEV FOR EACH BASKET AND POPULATION.
C
      INTEGER BSK
      COMMON /MOM/ BSKMN(10),BSKVAR(10),POPMN,POPVAR,NBASK1(10),POPTOT
      INTEGER BASKET,BASK1
      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
      INTEGER BIDNUM,BIDTYP,BIDOR
      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
      DO 1 BSK=1,NBASK
 1    BSKVAR(BSK)=0.
      POPMN=0.
      POPVAR=0.
      DO 3 BSK=1,NBASK
      NBID=0
      DO 2 N=1,NGROUP
      INDEX=BASK1(BSK,N)
      IF(INDEX.EQ.-1) GO TO 2
      NBID=NBID+1
      BSKVAR(BSK)=BSKVAR(BSK)+BIDPRI(INDEX)**2
 2    CONTINUE
      POPMN=POPMN+BTOT(BSK)
      POPVAR=POPVAR+BSKVAR(BSK)
      BSKMN(BSK)=BTOT(BSK)/NBID
      BSKVAR(BSK)=SQRT((BSKVAR(BSK)-BTOT(BSK)**2/NBID)/(NBID-1))
      NBASK1(BSK)=NBID
 3    CONTINUE
      POPVAR=SQRT((POPVAR-POPMN**2/NBIDS)/(NBIDS-1))
      POPTOT=POPMN
      POPMN=POPMN/NBIDS
      RETURN
      END
C
```

```
C
      SUBROUTINE OUTPUT
      CALL RNSTAT
      CALL LSTBSK
      CALL BSTATS
      RETURN
      END
C
C
      SUBROUTINE RNSTAT
C  PRINTS RUN STATISTICS.
C
      INTEGER BIDNUM,BIDTYP,BIDOR
      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
      INTEGER BASKET,BASK1
      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
      INTEGER CHRNAM,CHRNUM,CHRBSK
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRBSK(12,300)
      WRITE("     ",1)
    1 FORMAT('1',//,5X,'RUN STATISTICS')
      WRITE("     ",2) NBASK
    2 FORMAT(' ',/,10X,'NUMBER OF BASKETS =',8X,I3)
      WRITE("     ",4) NCHAR
    4 FORMAT(10X,'NUMBER OF CHARACTER TYPES =',I3)
      WRITE("     ",3) NBIDS
    3 FORMAT(10X,'NUMBER OF BIDS =',11X,I3)
      IF(NCHAR.EQ.1) GO TO 9
      WRITE("     ",5)
    5 FORMAT(10X,'NUMBER OF BIDS PER CHARACTER TYPE:',/)
      DO 6 N=1,NCHAR
    6 WRITE("     ",7) N,CHRNAM(N,1),CHRNAM(N,2),CHRNUM(N)
    7 FORMAT(20X,I2,'.',2X,2A4,2X,'=',2X,I3)
    9 WRITE("     ",8)
    8 FORMAT(' ',5X,40('-'))
      RETURN
      END
C
C
      SUBROUTINE LSTBSK
C  LISTS BASKETS.
C
      INTEGER BSK
      INTEGER BIDNUM,BIDTYP,BIDOR
      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
      INTEGER BASKET,BASK1
      COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
      INTEGER CHRNAM,CHRNUM,CHRBSK
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRBSK(12,300)
      DO 10 BSK=1,NBASK
      WRITE("     ",1) BSK
    1 FORMAT(' ',//,5X,'BASKET NO.',2X,I2)
```

```
      IF(NCHAR.GT.1) GO TO 3
      WRITE("     ",2)
 2    FORMAT(' ',/,12X,'PROPOSAL NO.',10X,'BID PRICE')
      GO TO 5
 3    WRITE("     ",4)
 4    FORMAT(' ',12X,'PROPOSAL NO.',10X,'BID PRICE',10X,'CHARACTERIST',
     &'IC',/)
      MINUS=0
 5    DO 8 N=1,NGROUP
      NN=N-MINUS
      INDEX=BASK1(BSK,N)
      IF(INDEX.EQ.-1) GO TO 9
      IF(NCHAR.EQ.1) GO TO 7
      WRITE("     ",6)NN,BIDNUM(INDEX),BIDPRI(INDEX),CHRNAM(BIDTYP(INDEX),1),
     &CHRNAM(BIDTYP(INDEX),2)
 6    FORMAT(T10,I3,'.',T17,I4,T35,F10.3,T53,2A4)
      GO TO 8
 9    MINUS=MINUS+1
      GO TO 8
 7    WRITE("     ",6) N,BIDNUM(INDEX),BIDPRI(INDEX)
 8    CONTINUE
10    CONTINUE
      WRITE("     ",11)
11    FORMAT(' ',5X,40('-'))
      RETURN
      END
C
C
      SUBROUTINE BSTATS
C   PRINTS BASKET STATISTICS.
C
      INTEGER BIDNUM,BIDTYP,BIDOR
      COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
      INTEGER BASKET,BASK1
      COMMON /BACK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
      INTEGER CHRNAM,CHRNUM,CHRBSK
      COMMON /CHRCTR/ NCHAR,CHRNAM(12,2),CHRNUM(12),CHRBSK(12,300)
      REAL LOWBND,MAX,MIN,MAXDIF
      COMMON /ERRBND/ MAXDIF,LOWBND,UPRBND
      COMMON /MOM/ BSKMN(10),BSKVAR(10),POPMN,POPVAR,NBASK1(10),POPTOT
      WRITE("     ",1)
 1    FORMAT(' ',/,5X,'BASKET STATISTICS',/)
      WRITE("     ",2) MAXDIF
 2    FORMAT(T11,'DIFFERENCE BETWEEN MAX AND MIN BASKETS =',2X,F10.3)
      WRITE("     ",3) LOWBND,UPRBND
 3    FORMAT(T11,'MEAN ERROR BOUNDED BY',F5.2,'% AND',F5.2,'%.')
      WRITE("     ",4)
 4    FORMAT(T11,'BID PRICE MOMENTS:')
      WRITE("     ",5)
 5    FORMAT(' ',/,T13,'BASKET',T23,'NO. PROPOSALS',T40,'TOTAL OF BIDS',
     &T60,'MEAN',T70,'STD DEV')
      DO 6 N=1,NBASK
 6    WRITE("     ",7) N,NBASK1(N),BTOT(N),BSKMN(N),BSKVAR(N)
 7    FORMAT(T14,I2,T29,I3,T42,F10.3,T55,F10.3,T66,F10.3)
      WRITE("     ",8) NBIDS,POPTOT,POPMN,POPVAR
 8    FORMAT(' ',T13,'POPUL',T29,I3,T42,F10.3,T55,F10.3,T66,F10.3)
      RETURN
      END
```

```
8630C
8640          SUBROUTINE CONT(*,*)
8650          INTEGER ANSWR,YES,NO
8660          DATA YES/'Y'/,NO/'N'/
8670          WRITE("     ",1)
8671 PRINT,""
8680     1    FORMAT( /,' DO YOU WISH TO SWAP BIDS BETWEEN SELECTED BASKETS?',
8690         &' (YES OR NO)')
8700     2    READ("     ",7) ANSWR
8710          IF(ANSWR.NE.YES) GO TO 3
8720          CALL SWAP2
8730          RETURN2
8740     3    IF(ANSWR.EQ.NO) GO TO 4
8750          WRITE("     ",9)
8751 PRINT,""
8760          GO TO 2
8770     4    WRITE("     ",5)
8771 PRINT,""
8780     5    FORMAT(' DO YOU WISH TO CHANGE THE NUMBER OF BASKETS? (YES OR NO)'
8790         &)
8800     6    READ("     ",7) ANSWR
8810     7    FORMAT(A1)
8820          IF(ANSWR.NE.YES) GO TO 8
8830          CALL NUMBSK
8840          RETURN1
8850     8    IF(ANSWR.EQ.NO) STOP
8860          WRITE("     ",9)
8861 PRINT,""
8870     9    FORMAT(' REENTER YES OR NO.')
8880          GO TO 6
8890          END
8900C
8910C
8920          SUBROUTINE SWAP2
8930C  PERFORMS MANUAL SWAPS.
8940C
8950          INTEGER BIDNUM,BIDTYP,BIDOR
8960          COMMON /BID/ NBIDS,BIDNUM(300),BIDPRI(300),BIDTYP(300),BIDOR(160)
8970          INTEGER BASKET,BASK1
8980          COMMON /BASK/ NBASK,BASKET(10,300),BASK1(10,160),BTOT(10),NGROUP
8990          INTEGER ANSWR,YES,NO,NBSK(2),NBID(2),IND(2),N1(2)
9000          DATA YES/'Y'/,NO/'N'/
9010      1   WRITE("     ",2)
9011 PRINT,""
9020      2   FORMAT(' INPUT: BASKET NUMBER, PROPOSAL NUMBER, BASKET NUMBER, ',
9030         &'PROPOSAL NUMBER.')
9040          READ("     ",*) NBSK(1),NBID(1),NBSK(2),NBID(2)
9050C
9060C  CHECK IF BASKET NUMBERS AND PROPOSAL NUMBERS ARE VALID.
9070C
```

```
0080          DO 6 I=1,2
0090          IF(1.LE.NBSK(I).AND.NBSK(I).LE.NBACK) GO TO 4
0100          WRITE("     ",3) NBSK(I)
0110       3  FORMAT(' *** ERROR ***',1X,I4,' IS NOT A VALID BASKET NUMBER.')
0120          GO TO 13
0130       4  IF(1.LE.NBID(I).AND.NBID(I).LE.NBIDS) GO TO 6
0140          WRITE("     ",5) NBID(I)
0150       5  FORMAT(' *** ERROR *** ',I5,' IS NOT A VALID PROPOSAL NUMBER.')
0160          GO TO 13
0170       6  CONTINUE
0180C
0190C  SEARCH BIDNUM(N) TO FIND APPROPRIATE INDICES.
0200C
0210          DO 8 I=1,2
0220          DO 7 INDEX=1,NBIDS
0230          IF(BIDNUM(INDEX).EQ.NBID(I)) GO TO 8
0240       7  CONTINUE
0250       8  IND(I)=INDEX
0260          DO 11 I=1,2
0270          DO 9 N=1,NGROUP
0280          IF(BASK1(NBSK(I),N).EQ.IND(I)) GO TO 11
0290       9  CONTINUE
0300          WRITE("     ",10) BIDNUM(IND(I)),NBSK(I)
0310      10  FORMAT(' *** ERROR ***  PROPOSAL ',I4,' IS NOT IN BASKET ',I3,'.')
0320          GO TO 13
0330      11  N1(I)=N
0340C
0350C  SWAP BIDS AND ADJUST BASKET TOTALS.
0360C
0370          DO 12 I=1,2
0380          BASK1(NBSK(I),N1(I))=IND(3-I)
0390      12  BTOT(NBSK(I))=BTOT(NBSK(I))-BIDPRI(IND(I))+BIDPRI(IND(3-I))
0400      13  WRITE("     ",14)
0401 PRINT,""
0410      14  FORMAT(' DO YOU WISH TO REENTER SWAP OR ENTER MORE SWAPS? ',
0420         &'(YES OR NO)')
0430      15  READ("    ",16) ANSWR
0440      16  FORMAT(A1)
0450          IF(ANSWR.EQ.YES) GO TO 1
0460          IF(ANSWR.EQ.NO) RETURN
0470          wRITE("     ",17)
0471 PRINT,""
0480      17  FORMAT(' REENTER YES OR NO.')
0490          GO TO 15
0500          END
```

[illegible]   1 :INPUT   [illegible]

```
      REAL BID(300),NEGOTD(150)
      INTEGER PROPSN(150),RETURN
      LOGICAL QUIT
      FILENAME BIDFIL,NEGFIL
      PRINT 10
   10 FORMAT("-",80("*")/" *",78(" "),"*"/" *",30X,"PRICE COMPUTATIONS",
  &   30X,"*"/" *",26X,"BASED ON NEGOTIATED SAMPLE",26X,"*"/" *",78X,
  &   "*"/" ",80("*")/" ")
      RETURN=2
   20 PRINT 30
   30 FORMAT("0ENTER NAME OF FILE CONTAINING PROPOSAL NUMBERS ",
  &   "AND NEGOTIATED PRICES ")
      READ 40,NEGFIL
OPEN(NEGFIL,ERR=110,STATUS="OLD")
      READ (NEGFIL,40) BIDFIL
   40 FORMAT(A8)
      RETURN=1
OPEN(BIDFIL,ERR=110,STATUS="OLD")
      DO 50 I=1,300
   50 READ (BIDFIL,*,END=70,ERR=110) BID(I)
      I=301
   70 NBID=I-1
CLOSE(BIDFIL)
      RETURN=2
      DO 100 I=1,150
  100 READ (NEGFIL,*,END=150,ERR=110) PROPSN(I),NEGOTD(I)
      I=151
      GO TO 150
110 I=IERROR()
      GO TO (120,140),RETURN
  120 PRINT 130,BIDFIL,I
  130 FORMAT("0ERROR ENCOUNTERED ON ",A8," FILE. FAILURE CODE IS",I4,".")
      GO TO 20
  140 PRINT 130,NEGFIL,I
      GO TO 20
  150 NNEG=I-1
CLOSE NEGFIL
      IORDER=NNEG
      DO 170 K=2,NNEG
      QUIT=.TRUE.
      DO 160 I=2,IORDER
      IM1=I-1
      IF (PROPSN(IM1).LE.PROPSN(I)) GO TO 160
      QUIT=.FALSE.
      IHOLD=PROPSN(I)
      PROPSN(I)=PROPSN(IM1)
      PROPSN(IM1)=IHOLD
      HOLD=NEGOTD(I)
      NEGOTD(I)=NEGOTD(IM1)
      NEGOTD(IM1)=HOLD
  130 CONTINUE
      IF (QUIT) GO TO 150
  140 IORDER=IORDER-1
  150 TNEG=0.
      TBIDS=0.
      DO 160 I=1,NNEG
      TNEG=TNEG+NEGOTD(I)
  160 TBIDS=TBIDS+BID(PROPSN(I))
      RATIO=TNEG/TBIDS
      PRINT 200
  200 FORMAT("-PROPOSAL",12X,"BID",6X,"ESTIMATED",5X,"NEGOTIATED"/
      2X,"NUMBER",3(10X,"PRICE")/" ")
      TOTBID=0.
      TOTEST=0.
      J=1
      DO 250 I=1,NBID
      EST=RATIO*BID(I)
      IF (PROPSN(J).EQ.I.AND.J.LE.NNEG) GO TO 220
      PRINT 210,I,BID(I),EST
  210 FORMAT(I9,F15.3,F15.3)
      TOTEST=TOTEST+EST
      GO TO 240
  220 PRINT 230,PROPSN(J),BID(I),NEGOTD(J)
  230 FORMAT(I9,F15.3,15X,F15.3)
      J=J+1
  240 TOTBID=TOTBID+BID(I)
  250 CONTINUE
      PRINT 260,TOTBID,TOTEST,TNEG
  260 FORMAT(9X,3(6X,9("-"))/"   TOTALS",3F15.3/"-")
      STOP
      END
```

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER N134	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Basket Method for Selecting Balanced Samples - Part III: Compuer Source Programs		5. TYPE OF REPORT & PERIOD COVERED Technical 6. PERFORMING ORG. REPORT NUMBER T.R. #374
7. AUTHOR(s) K.T. Wallenius Stephen L. Benz		8. CONTRACT OR GRANT NUMBER(s) N00014-75-C-0451
9. PERFORMING ORGANIZATION NAME AND ADDRESS Clemson University Dept. of Mathematical Sciences Clemson, South Carolina 29631		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NR 365-049
11. CONTROLLING OFFICE NAME AND ADDRESS Office of Naval Research Code 434 Arlington, Va. 22217		12. REPORT DATE 30 Nov. 1981 13. NUMBER OF PAGES 58
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Balanced sampling.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

In this third of a series of three documents describing the basket method of sampling, source computer codes are provided for the convenience of users who may want to modify the programs to meet special local needs.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

DATE
ILMED
8